



*Filtragem de ruído Speckle em imagens SAR com
redes neurais convolucionais e distâncias
estocásticas*

Pedro Henrique Salles Arouck de Souza

Janeiro / 2025

Dissertação de Mestrado em Ciência da
Computação

Filtragem de ruído Speckle em imagens SAR com redes neurais convolucionais e distâncias estocásticas

Esse documento corresponde à Dissertação apresentada à Banca Examinadora para Defesa de Dissertação no curso de Mestrado em Ciência da Computação da UNIFACCAMP – Centro Universitário Campo Limpo Paulista.

Campo Limpo Paulista, 3 de abril de 2025.

Pedro Henrique Salles Arouck de Souza

Prof. Dr. José Hiroki Saito (Orientador)

Este trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001.

**Ficha catalográfica elaborada pela
Biblioteca Central da Unifaccamp**

S717F

Souza, Pedro Henrique Salles Arouck de

Filtragem de ruído *Speckle* em imagens SAR com redes neurais convolucionais e distâncias estocásticas / Pedro Henrique Salles Arouck de Souza. Campo Limpo Paulista, SP: Unifaccamp, 2025.
71 p.: il.

Orientador: Prof. Dr. Jose Hiroki Saito

Dissertação (Programa de Mestrado Profissional em Ciência da Computação) – Centro Universitário Campo Limpo Paulista – Unifaccamp.

1. Redes neurais convolucionais. 2. Ruído. 3. *Speckle*. 4. Imagens SAR. 5. Processamento de imagens. 6. Distâncias estocásticas. I. Saito, Jose Hiroki. II. Centro Universitário Campo Limpo Paulista. III. Título.

CDD - 006.32

Dedicatória

Dedico este trabalho à memória do professor Nelson Mascarenhas, que me apresentou ao estudo de Processamento Digital de Imagens.

Agradecimentos

A conclusão deste trabalho não teria sido possível sem o apoio e a dedicação de muitas pessoas, às quais expresso minha profunda gratidão.

Agradeço à minha esposa, pelo companheirismo, paciência e incentivo em todos os momentos. Seu apoio foi fundamental para que eu chegasse até aqui.

À minha família, pelo suporte incondicional, pelos ensinamentos e pelo incentivo ao longo da minha trajetória acadêmica.

Aos meus colegas de turma, pelo compartilhamento de conhecimento, pelo incentivo e pelas trocas ao longo dessa jornada.

Aos professores que contribuíram para minha formação acadêmica e à equipe da UNIFACCAMP, pelo suporte e pelos ensinamentos ao longo do curso.

Um agradecimento especial ao professor José Hiroki Saito, por sua orientação e apoio fundamentais para que eu conseguisse completar este desafio.

A todos que fizeram parte desta caminhada, meu muito obrigado.

”O essencial é invisível aos olhos.”

— Antoine de Saint-Exupéry

Resumo

Esta dissertação apresenta um estudo sobre a filtragem de ruído speckle em imagens SAR utilizando redes neurais convolucionais. O objetivo é desenvolver uma solução acessível e eficiente para melhorar a qualidade das imagens degradadas por esse tipo de ruído, facilitando sua análise e interpretação. A metodologia adotada envolve a simulação de ruído speckle em imagens limpas, o treinamento de uma rede neural convolucional e a avaliação dos resultados por meio de métricas tradicionais, como PSNR e SSIM. Além disso, distâncias estocásticas foram agregadas à função de perda da rede neural convolucional, permitindo uma análise mais detalhada da preservação das propriedades estatísticas das imagens filtradas. O treinamento foi realizado com o banco de dados BSDS500, utilizando apenas hardware não especializado e ferramentas de código aberto. Os resultados indicam que o modelo é capaz de reduzir o ruído speckle, preservando detalhes estruturais da imagem. Além disso, a comparação com estudos recentes sobre o tema mostrou que a solução proposta apresenta desempenho competitivo. A implementação acessível e a viabilidade do uso de equipamentos não especializados reforçam o potencial do método para aplicações práticas e futuras pesquisas na área.

Palavras-chave: Redes Neurais Convolucionais, Ruído, Speckle, Imagens SAR, Processamento de Imagens, Distâncias Estocásticas

Abstract

This work presents a study on speckle noise filtering in SAR images using convolutional neural networks. The objective is to develop an accessible and efficient solution to improve the quality of images degraded by this type of noise, facilitating their analysis and interpretation. The proposed methodology involves the simulation of speckle noise in clean images, the training of a convolutional neural network, and the evaluation of results using traditional metrics such as PSNR and SSIM. Additionally, stochastic distances were integrated into the loss function of the convolutional neural network, enabling a more detailed analysis of the preservation of the statistical properties of the filtered images. The training was conducted using the BSDS500 dataset, utilizing only non-specialized hardware and open-source tools. The results indicate that the model effectively reduces speckle noise while preserving the structural details of the image. Furthermore, comparisons with recent studies on the topic showed that the proposed solution achieves competitive performance. The accessible implementation and the feasibility of using non-specialized hardware reinforce the potential of this method for practical applications and future research in the field.

Keywords: Convolutional Neural Networks, Noise, Speckle, SAR Images, Image Processing, Stochastic Distances

Sumário

1	Introdução	14
1.1	Motivação, Objetivos e Justificativa	16
1.2	Organização deste Trabalho	18
2	Fundamentação Teórica	19
2.1	Fundamentos de Imagem Digital	19
2.1.1	Formação de Imagens	20
2.2	Imagens SAR e o Ruído Speckle	21
2.2.1	Princípio de Funcionamento	22
2.2.2	Ruído Speckle	24
2.3	Métodos Tradicionais de Redução de Ruído	26
2.3.1	Filtro de Lee	26
2.3.2	Filtro de Frost	27
2.3.3	Filtro de Kuan	27
2.3.4	Filtro Gamma-MAP	27
2.4	Redes Neurais Convolucionais	28
2.4.1	Arquitetura e Funcionamento das CNNs	28
2.4.2	Aplicação de CNNs no Processamento de Imagens	30
2.5	Distribuição de Probabilidade Gama e Distâncias Estocásticas	31
2.5.1	Divergência de Kullback-Leibler (KL)	32
2.5.2	Divergência de Jensen-Shannon (JS)	32
2.5.3	Distância de Hellinger	32
2.5.4	Distância Bhattacharyya	33

2.5.5	Divergência de Rényi	33
2.5.6	Distância Triangular	33
2.5.7	Distância Harmônica Média	33
2.5.8	Distância Aritmético-Geométrica	34
2.6	Base de Dados BSDS500	34
2.7	Medidas de Avaliação da Qualidade de Imagens	35
2.8	Índices de Avaliação para Remoção de Speckle	36
2.9	Outros índices de qualidade	36
3	Revisão de literatura	38
3.1	Planejamento	38
3.2	Execução da Revisão	39
3.2.1	Busca e Seleção de Artigos	39
3.2.2	Filtragem por Relevância e Aplicação de Critérios	40
3.2.3	Análise dos Artigos Seleccionados	41
3.2.4	Avaliação Final e Seleção dos Artigos Definitivos	41
3.3	Resultados da revisão	42
3.4	Conclusão da Revisão	43
4	Metodologia	44
4.1	Preparação de Dados	44
4.1.1	Conjunto de Dados: BSDS500	45
4.1.2	Redimensionamento das Imagens	45
4.1.3	Adição de Ruído Speckle	45
4.1.4	Normalização das Imagens	46
4.1.5	Divisão em Conjuntos de Treinamento e Validação	46

4.1.6	Organização dos Dados para Treinamento	46
4.2	Desenvolvimento e Treinamento do Modelo	47
4.2.1	Desenvolvimento do Modelo	47
4.2.2	Treinamento do Modelo	47
4.3	Avaliação com Distâncias Estocásticas	49
4.4	Análise Qualitativa e Quantitativa	49
4.5	Implementação do Código	49
4.5.1	Bibliotecas Utilizadas no Projeto	50
4.6	Apresentação do Script	51
5	Experimentos, Resultados e Discussões para Validação	52
5.1	Preparação dos Dados	52
5.2	Treinamento do Modelo	52
5.3	Avaliação do Modelo	53
5.4	Configuração Experimental	53
5.4.1	Configurações de Hardware e Software	53
5.4.2	Configurações de Treinamento	54
5.5	Resultados Quantitativos	54
5.6	Resultados Qualitativos	56
5.6.1	Imagem Original	56
5.6.2	Imagens Contaminadas com Ruído Speckle	57
5.6.3	Resultados de Filtragem	58
5.6.4	Análise Visual	58
5.7	Discussão dos Resultados	59
5.7.1	Impacto dos Parâmetros no Desempenho	59

5.7.2	Comparação com Trabalhos Relacionados	60
5.7.3	Limitações do Modelo e Considerações Finais	60
6	Conclusão	62
6.1	Trabalhos Futuros	63
	Referências	64
	Apêndice A – Apresentação do script	68

Lista de acrónimos e abreviaturas

BSDS500	Berkeley Segmentation Dataset and Benchmark
CNN	Convolutional Neural Network
InSAR	Radar Interferométrico de Abertura Sintética
LReLU	Leaky Rectified Linear Activation
MAP	Maximum A Posteriori
MSE	Mean Squared Error
NLM	Non-Local Means
PSNR	Peak Signal to Noise Ratio
ReLU	Rectified Linear Unit
RGB	Red, Green and Blue
SAR	Synthetic Aperture Radar
SI	Speckle Index
SMPI	Suppression and Mean Preservation Index
SSI	Speckle Suppression Index
SSIM	Structural Similarity Index Measure

Lista de Tabelas

1	Critérios de Inclusão	39
2	Critérios de Exclusão	40
3	Resultados da Busca e Seleção de Artigos	40
4	Resultados após Filtragem por Relevância e Aplicação de Critérios	41
5	Resultados após Análise dos Artigos Seleccionados	41
6	Resultados Finais após Avaliação e Seleção dos Artigos Definitivos . . .	41
7	Artigos Seleccionados sobre Filtragem de Speckle em Imagens SAR . . .	43
8	Configurações de Hardware e Software	54
9	Configurações de Treinamento	54
10	Resultados quantitativos para <i>Looks</i> = 1	55
11	Resultados quantitativos para <i>Looks</i> = 5	55

Lista de Figuras

1	Comparação óptica/radar da erupção do vulcão Kliuchevskol: (a) óptica e (b) radar.	22
2	Sistema básico de sensoriamento remoto	23
3	Representação visual do ruído speckle	26
4	Arquitetura básica de uma rede neural convolucional	29
5	Imagem sem ruído e normalizada, utilizada como referência.	57
6	À esquerda: Imagem contaminada com ruído speckle, $L = 1$. À direita: Imagem contaminada com ruído speckle, $L = 5$	57
7	Resultados de filtragem. Da esquerda para a direita: (1) $L = 1$, 20 imagens, 1 época; (2) $L = 5$, 100 imagens, 10 épocas; (3) $L = 5$, 200 imagens, 100 épocas.	58

1. Introdução

A filtragem de imagens é uma técnica amplamente utilizada em diversas áreas, como Astronomia, Medicina, Ciência Forense, Engenharias, entre outras. Na área de Processamento de Imagens e Sinais, as técnicas para a atenuação ou remoção de ruídos são continuamente estudadas por pesquisadores (Tian et al. 2020). Essas técnicas procuram recuperar imagens corrompidas com base em um conhecimento a priori do fenômeno de degradação, além de desenvolver critérios para definir estimativas do resultado desejado e avaliar a qualidade da imagem filtrada (Gonzalez & Woods 2017).

Várias técnicas têm sido propostas para reduzir o ruído speckle. Sendo assim, a partir da década de 1980, os primeiros filtros surgiram sob o critério de erro quadrático médio e utilizando a vizinhança espacial dos pixels, como os de Lee (1980), Kuan et al. (1985) e Frost et al. (1982). Na década de 1990, foram propostos filtros baseados no critério de variação total (Rudin et al. 1992) e wavelets (Donoho & Johnstone 1995). A partir dos anos 2000, surgiram os chamados filtros de média não local (em inglês, *non-local means*, NLM), dominando o cenário de filtragem de ruído (Buades et al. 2005). Mais recentemente, filtros baseados em aprendizado profundo emergiram com resultados promissores, sendo que tais filtros requerem um grande volume de dados para seu treinamento (Fracastoro et al. 2021).

Os sistemas de SAR (em inglês, *Synthetic Aperture Radar*) são extremamente úteis para monitorar a superfície da Terra, com aplicações que incluem a detecção de vazamentos de óleo nos oceanos, monitoramento de geleiras derretidas e análise de fronteiras. A vantagem destes sistemas é a sua aplicabilidade em variadas condições climáticas, pois operam na faixa de micro-ondas, tornando-os menos afetados que os sensores ópticos (Penna & Mascarenhas 2018).

As imagens SAR são frequentemente afetadas pelo ruído speckle, decorrente da natureza coerente (uso de ondas eletromagnéticas com uma relação bem definida de fase e frequência) do imageamento SAR. Esse fenômeno ocorre quando o sistema emite pulsos eletromagnéticos em direção a um alvo e analisa o eco de retorno. Durante o processamento coerente, interferências causadas pelos retroespalhamentos na superfície geram o ruído speckle, que degrada a qualidade visual, dificultando análises e interpretações

precisas, além de tornar essas tarefas mais complexas e suscetíveis a erros (Penna & Mascarenhas 2019).

Este ruído é causado pela característica coerente do sensor, pois a amplitude e fase do sinal de retorno são resultantes de uma soma vetorial dos sinais de retorno de vários difusores contidos em cada célula de resolução. A fase de cada difusor está relacionada à distância entre o difusor e o sensor. Se o sensor se move, as fases dos difusores mudarão, acarretando uma alteração na amplitude total, que se manifesta como um padrão granular nas imagens (Dutra et al. 2003).

No entanto, o speckle não é apenas ruído, pois ele carrega informações sobre o alvo processado. Portanto, para um método de redução de speckle ser eficaz, ele deve remover o speckle enquanto preserva os detalhes da imagem, como bordas, linhas e pontos (Jansing 2021). Além disso, o speckle corrompe o sinal de retorno de forma multiplicativa, já intensidade do ruído em cada ponto da imagem depende proporcionalmente do valor do sinal original nesse ponto. Portanto, a distribuição de ruído não segue o padrão Gaussiano, o ruído speckle segue uma distribuição Gamma (Penna & Mascarenhas 2018).

Em anos recentes, grande atenção tem sido dada aos filtros de média não-locais, nos quais a estimativa de um pixel utiliza informações de todos os outros pixels da imagem, em vez de considerar apenas os pixels vizinhos, como nas abordagens locais (Buades et al. 2005). Uma característica fundamental de qualquer técnica não-local é a existência de uma medida de similaridade entre *patches*, que são pequenos blocos ou regiões retangulares extraídas da imagem para análise. Essa similaridade é tipicamente medida pela distância Euclidiana em casos de ruído aditivo gaussiano. No filtro non-local means, proposto por Buades et al. (2005), essa medida de similaridade é utilizada para calcular os pesos de uma média ponderada aplicada durante o processo de filtragem.

Mais recentemente, os avanços em aprendizado profundo têm proporcionado resultados promissores na filtragem de speckle (Tian et al. 2020). Redes neurais convolucionais (em inglês, *Convolutional Neural Network* - CNN) têm se mostrado particularmente eficazes, mas enfrentam desafios relacionados à exigência de grandes volumes de dados para treinamento. Comparados aos métodos tradicionais, os filtros baseados em aprendizado profundo podem oferecer uma maior capacidade de generalização e melhor

desempenho em diferentes condições de imagem, mas também, demandam maior poder computacional e dados de treinamento anotados (Fracastoro et al. 2021).

Ao longo deste trabalho, busca-se demonstrar como a integração de técnicas de aprendizado profundo e distâncias estocásticas pode oferecer uma solução robusta e eficaz para a mitigação do ruído speckle, contribuindo para o avanço na interpretação e análise de imagens SAR. Os objetivos específicos incluem o desenvolvimento de um novo filtro de speckle, a avaliação comparativa com métodos tradicionais e baseados em redes convolucionais e a aplicação prática dos resultados em cenários reais de uso das imagens SAR.

Com esta abordagem, espera-se não apenas aprimorar as técnicas de filtragem de speckle, mas também abrir caminhos para pesquisas futuras na área de processamento de imagens SAR, com impactos positivos em diversas aplicações científicas e tecnológicas.

1.1. Motivação, Objetivos e Justificativa

A filtragem de ruído speckle em imagens obtidas por sensores SAR é um tema de pesquisa altamente relevante e dinâmico na área de processamento de imagens. Essas imagens são amplamente utilizadas em diversas aplicações, como monitoramento ambiental, detecção de desastres naturais, vigilância e mapeamento geográfico. A capacidade de operar em todas as condições meteorológicas e de penetração de nuvens torna o SAR uma ferramenta indispensável para a observação da Terra. No entanto, a presença do ruído speckle pode reduzir severamente o desempenho de técnicas automatizadas de análise de cenas e extração de informações, além de ser prejudicial em aplicações que exigem múltiplas observações SAR, como na detecção automática de mudanças multitemporais (Argenti et al. 2013).

Segundo Fracastoro et al. (2021), o uso de técnicas de aprendizado profundo para *despeckling* é uma área de pesquisa atual, com um grande volume de estudos sendo conduzidos e muitas questões ainda a serem exploradas. Este campo do conhecimento tem atraído a atenção de pesquisadores que buscam soluções mais eficazes e eficientes para o problema do speckle.

O objetivo principal desta dissertação é desenvolver um método inovador para a

filtragem de ruído speckle em imagens SAR, combinando redes neurais convolucionais com distâncias estocásticas. Os objetivos específicos incluem:

- 1.Desenvolver uma abordagem baseada em CNN para a filtragem de speckle em imagens SAR;
- 2.Integrar distâncias estocásticas na modelagem do filtro, visando melhorar a robustez e a eficiência computacional do método proposto;
- 3.Avaliar o desempenho do método proposto em comparação com filtros tradicionais e outros métodos modernos de remoção de ruído speckling, utilizando métricas de qualidade de imagem;
- 4.Proporcionar uma base teórica sólida para a compreensão e desenvolvimento de futuros filtros de speckle, contribuindo para o avanço do estado da arte na área.

A justificativa para este trabalho reside na necessidade contínua de aprimoramento das técnicas de filtragem de speckle em imagens SAR. A qualidade das imagens é crucial para aplicações críticas que demandam alta precisão e confiabilidade, como monitoramento ambiental, vigilância e resposta a desastres naturais. O ruído speckle, ao introduzir incertezas e dificultar a interpretação das imagens, pode comprometer significativamente essas aplicações.

Métodos de filtragem baseados em aprendizado profundo, como as redes neurais convolucionais, apresentam um potencial significativo para melhorias, mas enfrentam desafios relacionados à necessidade de grandes volumes de dados para treinamento e à formalização matemática robusta exigida por algoritmos mais avançados. Modelos de maior desempenho frequentemente dependem de estruturas algorítmicas complexas, o que demanda explicações matemáticas mais detalhadas. No entanto, há pouca ou nenhuma evidência empírica de que um aumento na explicabilidade matemática dos algoritmos tenha impacto direto para o usuário final (Herm et al. 2023).

A pesquisa em *despeckling* utilizando aprendizado profundo é uma área em crescimento, com muitas oportunidades para inovação e descobertas. Este trabalho se insere nesse contexto, buscando contribuir com uma abordagem que não apenas melhore a eficiência e a precisão da filtragem de speckle, mas também forneça uma base teórica para futuros desenvolvimentos.

1.2. Organização deste Trabalho

Este trabalho está organizado da seguinte forma: após este primeiro capítulo, que apresenta a introdução, o Capítulo 2 aborda os fundamentos de imagens digitais e os principais métodos de filtragem de speckle desenvolvidos ao longo das últimas décadas. Em seguida, no Capítulo 3, discute-se a revisão de literatura sobre a abordagem baseada em aprendizado profundo para a redução do ruído speckle em imagens SAR. No Capítulo 4, é apresentada a metodologia proposta, que trata da preparação dos dados e do modelo, bem como das métricas a serem calculadas. No Capítulo 5, o experimento é apresentado, bem como os resultados, seguidos de uma discussão sobre as descobertas. No Capítulo 6, apresenta-se a conclusão, juntamente com as próximas etapas a serem tomadas para a composição da dissertação.

2. Fundamentação Teórica

2.1. Fundamentos de Imagem Digital

Gonzalez & Woods (2017) define uma imagem digital como a representação digital de uma cena visual capturada por um dispositivo sensor, como câmeras ópticas, sensores de radar ou dispositivos médicos.

Ela é formada por uma matriz bidimensional de elementos finitos chamados pixels, cada pixel de uma imagem é associado a um valor de intensidade, que representa a quantidade de luz capturada ou sintetizada naquele ponto específico. Em imagens *grayscale* (ou de escala de cinza), os valores dos pixels variam em uma faixa contínua que representa as diferentes intensidades de brilho. Por exemplo, em uma imagem de 8 bits, cada pixel pode assumir valores entre 0 (preto) e 255 (branco), com tons de cinza intermediários. Este tipo de codificação é chamado de profundidade de bits e define a faixa de valores de intensidade que um pixel pode representar.

Em imagens coloridas, cada pixel é representado por um conjunto de valores que correspondem a diferentes componentes de cor, geralmente no espaço de cores RGB (vermelho, verde e azul). Cada componente de cor é armazenado separadamente em uma matriz correspondente, e a combinação desses três valores determina a cor final do pixel na imagem. Imagens com maior profundidade de cor, como 24 bits (8 bits por canal), são capazes de representar milhões de cores diferentes, tornando-se adequadas para aplicações que exigem alta fidelidade visual.

A resolução espacial de uma imagem refere-se ao número de pixels por unidade de área, o que determina o nível de detalhe que pode ser percebido. Imagens com uma maior resolução possuem um número maior de pixels, permitindo uma representação mais detalhada da cena. A profundidade de cor também desempenha um papel importante na representação de imagens digitais. Ela define o número de bits usados para representar a intensidade de cada pixel, o que impacta diretamente a precisão das informações de cor ou brilho armazenadas.

2.1.1. Formação de Imagens

A formação de imagens é o processo pelo qual uma cena ou objeto do mundo real é capturado e convertido em uma representação digital para posterior processamento e análise. Dependendo da natureza da fonte de imagem, diferentes sensores e técnicas são utilizados para adquirir as imagens, cada um com suas próprias características e aplicações.

Imagens ópticas são formadas através da captura da luz visível refletida ou emitida por objetos. Câmeras digitais, por exemplo, utilizam sensores para converter os fótons de luz em sinais elétricos que podem ser digitalizados e armazenados. O processo de formação de imagem em câmeras ópticas envolve a passagem da luz através de lentes, que focalizam a cena em uma matriz de pixels no sensor, onde cada pixel gera um valor de intensidade correspondente à luz recebida.

A qualidade da imagem óptica depende de fatores como a iluminação da cena, a resolução espacial do sensor e a abertura da lente, que controla a quantidade de luz que entra na câmera. Em imagens médicas ou de alta precisão, como as utilizadas em astronomia, sensores especializados são usados para capturar detalhes sutis, com ênfase em melhorar a sensibilidade e a resolução (Gonzalez & Woods 2017).

Imagens SAR são obtidas por sensores de radar que operam na faixa de micro-ondas do espectro eletromagnético. Diferente de câmeras ópticas, que dependem da luz visível, sistemas SAR emitem pulsos de rádio em direção a um alvo e analisam o eco de retorno desses pulsos, medindo a intensidade do sinal retroespalhado pela superfície da Terra. A formação de uma imagem SAR envolve o processamento desse sinal em uma matriz de dados complexos, onde cada pixel contém informações de magnitude e fase (Penna & Mascarenhas 2019).

A principal vantagem do SAR é sua capacidade de adquirir imagens sob quaisquer condições meteorológicas e de iluminação, pois as ondas de radar podem penetrar através de nuvens, chuva e nevoeiro, além de serem independentes da luz solar. Isso faz com que os sistemas SAR sejam amplamente utilizados em monitoramento ambiental, vigilância militar e análise de mudanças no uso da terra (Podest 2024). Por exemplo, radares SAR têm sido utilizados para o monitoramento de florestas, detecção de desastres naturais e

estudo de movimentação de geleiras (Dalsasso et al. 2020).

Outra importante categoria refere-se ao imageamento médico, onde diferentes modalidades de imagem são usadas para capturar informações do corpo humano para diagnóstico e tratamento. Técnicas como Tomografia Computadorizada, Ressonância Magnética e Ultrassom utilizam diferentes fontes de energia para gerar imagens (Nath 2013).

Esses sistemas seguem o mesmo princípio básico de formação de imagens: a aquisição de dados a partir de uma fonte de energia, a interação desses dados com o objeto de interesse (seja através de retroespalhamento ou absorção), e a digitalização final desses sinais em uma matriz de valores digitais que representa a imagem. A forma como uma imagem é capturada e formada impacta diretamente os algoritmos utilizados no processamento subsequente.

De acordo com Gonzalez & Woods (2017), o processamento de imagens digitais envolve a manipulação de dados de imagens para aprimorar sua qualidade ou extrair informações úteis. O objetivo principal dessas técnicas é transformar ou melhorar a imagem de forma a torná-la mais adequada para análise visual ou para outras operações computacionais.

O conhecimento sobre o processo de formação de imagens também permite o desenvolvimento de melhores técnicas de reconstrução, onde a qualidade da imagem pode ser aprimorada através de novos algoritmos de pós-processamento (Archana & Jeevaraj 2024).

O ruído em imagens digitais é um fenômeno que afeta a qualidade das imagens, tornando a interpretação visual mais difícil e impactando negativamente a precisão das análises automatizadas. O ruído interfere no processo de captura de imagens devido a vários fatores como limitações do sensor, condições de iluminação e características do ambiente (Gonzalez & Woods 2017).

2.2. Imagens SAR e o Ruído Speckle

O Radar de Abertura Sintética é uma tecnologia de sensoriamento remoto que permite a aquisição de imagens de alta resolução da superfície terrestre. A principal

vantagem do SAR em relação a sensores ópticos tradicionais é sua capacidade de operar em condições adversas, como pouca luz ou condições meteorológicas adversas, incluindo nuvens e tempestades, já que o radar utiliza micro-ondas, que não são bloqueadas por essas condições atmosféricas (Podest 2024).

A Figura 1 mostra a comparação de imagens ópticas (a) e de radar (b) do vulcão Kliuchevskoi em Kamchatka, na Rússia, que começou a entrar em erupção em 30 de setembro de 1994.

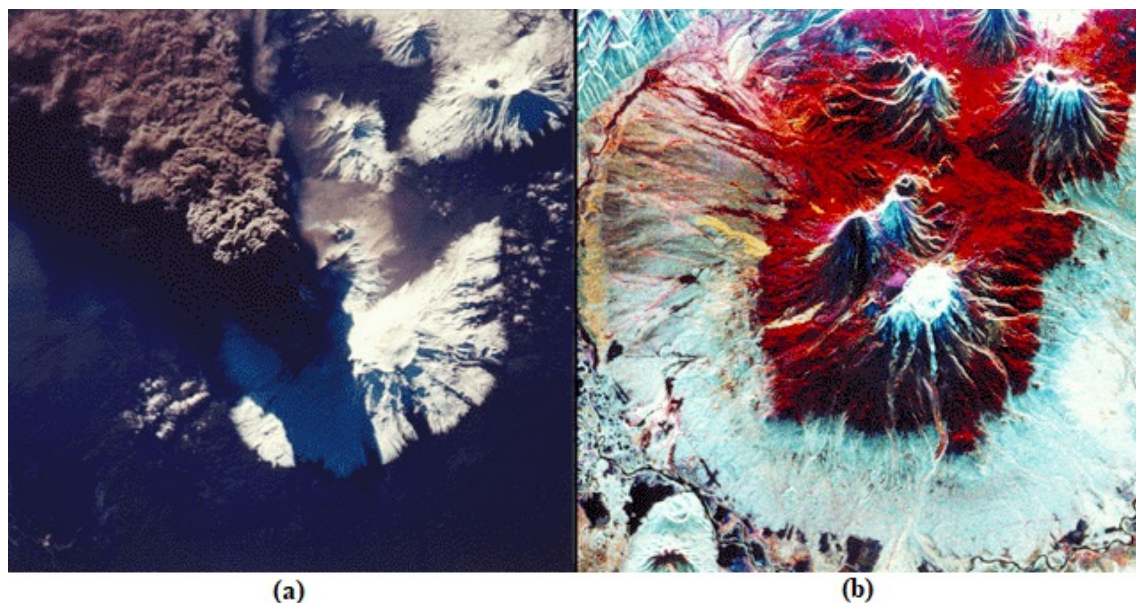


Figura 1. Comparação óptica/radar da erupção do vulcão Kliuchevskoi: (a) óptica e (b) radar. (Michigan Tech Volcanology 1994)

2.2.1. Princípio de Funcionamento

O Radar de Abertura Sintética funciona emitindo pulsos em direção à superfície terrestre e registrando os sinais que são refletidos de volta. O sistema é montado geralmente em uma plataforma móvel, como satélites ou aviões. À medida que a plataforma se move, o radar coleta dados a partir de várias posições ao longo da sua trajetória. A abertura sintética refere-se à combinação dos dados de múltiplas posições para simular uma antena de radar muito maior, o que melhora a resolução espacial da imagem (Podest 2024).

Na Figura 2, tem ilustrado um sistema básico de sensoriamento remoto onde é possível verificar: (a) uma fonte emissora de microondas; (b) o objeto alvo de espalha-

mento; (c) ondas emitidas; (d) ondas propagadas; (e) abertura coletora; (f) detector.

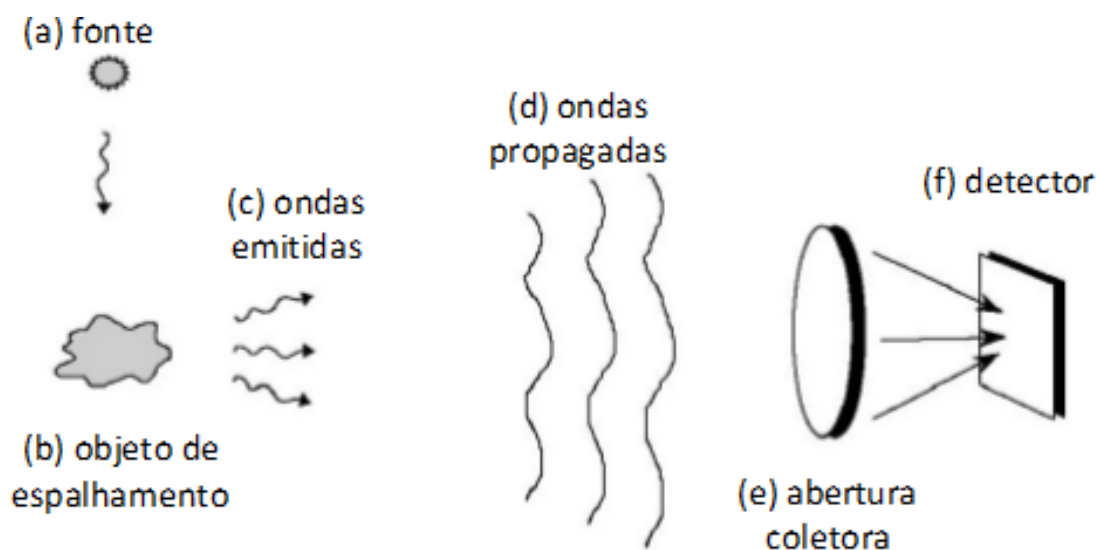


Figura 2. Sistema básico de sensoriamento remoto (Podest 2024)

Uma característica importante do SAR é que ele coleta tanto a amplitude quanto a fase do sinal de retorno. A amplitude está relacionada à intensidade da reflexão do sinal de radar, enquanto a fase contém informações sobre a distância e a elevação do terreno. Essa combinação de amplitude e fase permite a construção de modelos tridimensionais da superfície terrestre, uma técnica conhecida como interferometria SAR (InSAR), que é amplamente usada no monitoramento de deslocamentos de superfície, como deslizamentos de terra e terremotos (Fielding 2017).

O SAR é amplamente utilizado em uma variedade de aplicações devido à sua versatilidade e à alta resolução obtida nas imagens (Dalsasso et al. 2020). Algumas das principais aplicações incluem (Passah et al. 2021):

1. Monitoramento Ambiental: Imagens SAR são amplamente usadas para monitorar mudanças na cobertura do solo, como desmatamento e variações em ecossistemas aquáticos, bem como para medir a umidade do solo;
2. Vigilância e Segurança: Graças à sua capacidade de fornecer imagens de alta resolução independentemente do clima ou da luz, o SAR é uma ferramenta valiosa para a vigilância de fronteiras, monitoramento de áreas marítimas e detecção de atividades ilegais;
3. Geologia e Mapeamento: amplamente utilizado no mapeamento de características

geológicas, na detecção de falhas e na exploração de minerais. A interferometria SAR também tem sido fundamental no monitoramento de deformações superficiais;

4.Agricultura: Sensores SAR são utilizados para monitorar o desenvolvimento de colheitas, estimar a biomassa e detectar a umidade do solo em grandes áreas agrícolas, ajudando na gestão e planejamento agrícola.

Apesar das vantagens, as imagens SAR apresentam desafios de interpretação devido à sua natureza baseada em micro-ondas e à presença de efeitos de speckle, um tipo específico de ruído que degrada a qualidade da imagem. O efeito de *layover* (quando sinais de radar retornam em diferentes tempos devido a superfícies inclinadas) e a sombra também são desafios comuns, principalmente em terrenos montanhosos (Vitale et al. 2021).

2.2.2. Ruído Speckle

O ruído speckle, um tipo de ruído multiplicativo, ocorre nas imagens SAR devido à interferência coerente entre múltiplas ondas de radar refletidas de diferentes partes de uma superfície, causando um padrão granulado perceptível, especialmente em superfícies homogêneas como corpos d'água ou desertos. Esse fenômeno é resultado da natureza coerente do sinal de radar, onde cada pixel na imagem SAR representa a soma vetorial das reflexões de pequenas áreas dentro do objeto, chamadas de elementos de resolução. Ao refletirem o sinal de radar, esses objetos retornam ondas com fases ligeiramente diferentes, gerando interferência construtiva e destrutiva, o que provoca variações de intensidade no pixel e cria o padrão característico do speckle (Penna & Mascarenhas 2019).

O speckle tem uma natureza multiplicativa, o que significa que seu impacto é proporcional à intensidade do sinal refletido. Em áreas onde o retorno de sinal é mais forte, o nível de speckle é mais pronunciado (Passah et al. 2021).

No modelo multiplicativo de ruído speckle, o retorno y pode ser tratado como o produto entre duas variáveis aleatórias independentes: o speckle n e retroespalhamento do terreno x , caracterizado pela Equação 1 a seguir:

$$y = n \times x \quad (1)$$

O ruído speckle em imagens SAR segue uma distribuição Gama, cuja função densidade de probabilidade é definida como:

$$\rho(n) = \frac{L^L n^{L-1} \exp(-nL)}{\Gamma(L)} \quad (2)$$

onde:

- $\rho(n)$: Representa a função densidade de probabilidade (PDF) do ruído speckle, indicando a probabilidade de o ruído assumir um valor específico n .
- n : É o componente ruidoso, que varia e representa a intensidade do ruído multiplicativo em um pixel da imagem SAR.
- L : É o número de *looks*, que define a suavização do ruído speckle. Quanto maior o valor de L , menor a variabilidade do ruído (menor granularidade).
- $\Gamma(\cdot)$: É a função gama matemática.

O número de *looks* (L) é uma métrica que reflete o nível de integração de sinais na formação de uma imagem SAR. Ele está relacionado ao número de vezes que o radar observa a mesma área e combina essas observações para reduzir o ruído speckle.

Por fim, o número de *looks* está diretamente relacionado à variância do ruído speckle, que é inversamente proporcional a L , ou seja, um maior L reduz a variância do ruído, facilitando a análise visual e automatizada da imagem (Podest 2024).

A Figura 3 mostra uma representação visual da Equação 1. Pode-se observar que a imagem ruidosa y resulta da multiplicação do sinal de ruído n por uma imagem sem ruído x .

Com o avanço de técnicas de aprendizado profundo e modelos probabilísticos, novas abordagens têm surgido para a redução do speckle. Esses métodos utilizam modelos gerativos e redes neurais convolucionais para criar representações aprimoradas da imagem sem speckle (Fracastoro et al. 2021).

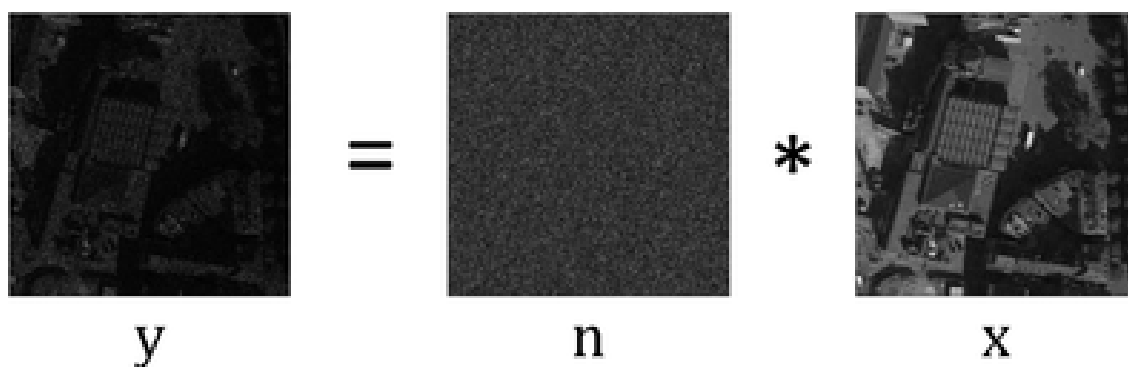


Figura 3. Representação visual do ruído speckle (Passah et al. 2021)

2.3. Métodos Tradicionais de Redução de Ruído

A remoção de ruído speckle em imagens SAR é um desafio significativo no processamento de imagens devido à natureza multiplicativa do ruído. Os métodos tradicionais de redução de ruído buscam melhorar a qualidade da imagem preservando detalhes essenciais, como bordas e texturas. Entre os métodos mais utilizados estão os filtros espaciais e estatísticos, que aplicam diferentes abordagens para reduzir o impacto do speckle. Aqui, discutiremos os principais filtros: Lee, Frost, Kuan e Gamma-MAP.

Com o surgimento de novas abordagens, como redes neurais convolucionais, esses métodos tradicionais podem ser combinados com essas novas para alcançar resultados ainda mais eficazes na remoção de speckle sem sacrificar detalhes críticos da imagem.

2.3.1. Filtro de Lee

O filtro de Lee é um dos métodos clássicos mais utilizados para a redução de ruído speckle em imagens SAR. Ele se baseia em uma abordagem estatística adaptativa, onde as características locais da imagem, como a variância e a média de uma janela de pixels, são usadas para ajustar a suavização aplicada a cada pixel (Argenti et al. 2013).

Lee (1980) utiliza a razão de máxima verossimilhança para suavizar o speckle de forma adaptativa. Ele opera assumindo que a intensidade de cada pixel é composta pela intensidade média local e pelo ruído multiplicativo. Para cada pixel dentro de uma janela deslizante, o filtro calcula a variância local e, com base nisso, determina um fator de suavização. Quanto maior for a variância local (indicando uma possível borda ou detalhe), menor será o efeito de suavização, permitindo a preservação de detalhes importantes. Em

áreas homogêneas, onde a variação é menor, o filtro aplica uma suavização mais intensa, reduzindo significativamente o speckle.

2.3.2. Filtro de Frost

O filtro de Frost (Frost et al. 1982) também é um método amplamente utilizado para a redução de speckle em imagens SAR, com uma abordagem baseada em convolução ponderada por um fator exponencial que considera a distância do centro da janela de filtragem. Ele é considerado um filtro espacial adaptativo, semelhante ao filtro de Lee, mas com um esquema de suavização diferente (Argenti et al. 2013).

Frost et al. (1982) utilizam uma função exponencial para suavizar os pixels, onde o peso atribuído a cada pixel dentro da janela depende da distância ao pixel central e da variância local. Esse peso decai exponencialmente conforme a distância ao pixel central aumenta, o que permite uma suavização mais intensa em áreas homogêneas e preserva as bordas em regiões de alta variação.

2.3.3. Filtro de Kuan

O filtro de Kuan é outro método clássico de redução de speckle que utiliza uma abordagem de filtragem baseada em estatísticas locais. Ele se diferencia do filtro de Lee por seu modelo matemático, que emprega uma estimativa de mínimos quadrados para suavizar o ruído multiplicativo (Argenti et al. 2013).

Kuan et al. (1985) modela o speckle como um processo multiplicativo e usa uma abordagem de mínimos quadrados para estimar a intensidade real dos pixels. Ele assume que a intensidade observada em cada pixel é composta pela intensidade verdadeira e pelo ruído multiplicativo. Assim, a suavização aplicada pelo filtro é proporcional à variância local, de forma semelhante ao filtro de Lee, mas com uma formulação mais robusta para lidar com diferentes distribuições de ruído.

2.3.4. Filtro Gamma-MAP

Com o avanço das técnicas de processamento de imagens, novos métodos tradicionais foram desenvolvidos para enfrentar os desafios apresentados pelo ruído speckle em imagens SAR. Um desses métodos mais recentes é o Filtro Gamma-MAP (Gamma

Maximum A Posteriori), que incorpora princípios estatísticos avançados para melhorar a redução do speckle mantendo a integridade estrutural da imagem (Argenti et al. 2013).

O Filtro Gamma-MAP baseia-se na modelagem estatística do speckle usando a distribuição Gamma, que é adequada para representar o ruído multiplicativo observado em imagens SAR. Este método estima a intensidade verdadeira de cada pixel utilizando uma abordagem de máxima a posteriori, onde a distribuição a priori das intensidades é combinada com a distribuição do ruído speckle para obter uma estimativa mais precisa (Lopes et al. 1990).

2.4. Redes Neurais Convolucionais

As Redes Neurais Convolucionais são uma das arquiteturas muito utilizadas no campo do aprendizado profundo, com aplicações que abrangem reconhecimento de imagens, visão computacional, e mais recentemente, tarefas de processamento de imagens, incluindo a redução de ruído, como o ruído speckle em imagens SAR (Liu & Liu 2022). Nesta seção, vamos examinar o funcionamento das CNNs e suas aplicações no processamento de imagens, destacando as últimas inovações.

2.4.1. Arquitetura e Funcionamento das CNNs

A arquitetura de uma CNN é composta de múltiplas camadas organizadas de forma hierárquica, onde cada camada realiza operações específicas, como convoluções, pooling e ativações, o que permite à rede aprender e capturar características complexas de uma imagem. As convoluções aplicam filtros para extrair características locais, como bordas e texturas. O pooling reduz a dimensão dos mapas de características, resumindo informações em pequenas regiões, enquanto as funções de ativação, como a ReLU (Rectified Linear Unit), introduzem não-linearidade, permitindo à rede aprender relações complexas (Passah et al. 2021).

A descrição seguinte refere-se aos elementos que constituem as camadas de uma CNN, rotulada na parte superior da Figura 4, da esquerda para a direita, excluindo a primeira camada de entrada (INPUT) e a última, de saída (OUTPUT).

A camada convolucional (CONV) é a unidade central da CNN, responsável pela extração de características. Ela aplica filtros ou *kernels* em toda a imagem de entrada, re-

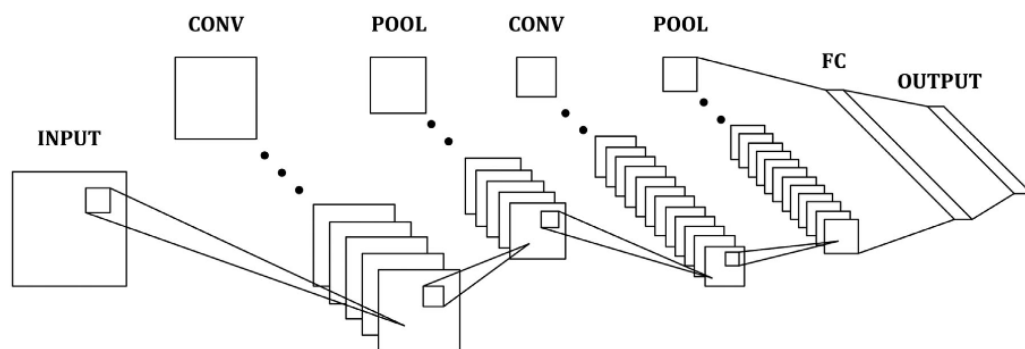


Figura 4. Arquitetura básica de uma rede neural convolucional (Passah et al. 2021)

alizando operações de convolução. Cada filtro convolui sobre a imagem para gerar mapas de características, que representam a resposta do filtro aplicado em diferentes partes da imagem. Diferentes filtros podem capturar diferentes aspectos da imagem, como bordas, texturas ou padrões geométricos (Passah et al. 2021).

O aprendizado dos pesos dos filtros é um dos grandes trunfos das CNNs. Durante o treinamento da rede, os pesos dos filtros são ajustados de maneira a melhorar o desempenho em uma tarefa específica, como a classificação ou redução de ruído. Isso contrasta com técnicas tradicionais de processamento de imagens, nas quais os filtros são projetados manualmente, como no caso dos filtros gaussianos ou de Sobel.

A camada de *pooling* (POOL) reduz o tamanho dos mapas de características, o que ajuda a reduzir a complexidade computacional e a aumentar a robustez da rede em relação a pequenas variações na imagem. A operação de *pooling* mais comum é o *max pooling*, que seleciona o valor máximo em uma pequena janela de pixels, permitindo que a rede se concentre nas características mais fortes de cada região. Outro tipo é o *average pooling*, que tira a média dos valores, proporcionando uma versão suavizada da imagem (Passah et al. 2021).

Essas camadas atuam como um "resumo" das informações da imagem, preservando apenas as características mais relevantes. Elas também ajudam a controlar o *overfitting*, reduzindo o número de parâmetros que a rede precisa aprender, o que é crucial em problemas de imagens grandes e complexas, como aquelas obtidas por SAR.

Depois de passarem pelas camadas convolucionais e de *pooling*, as características

extraídas da imagem são processadas por camadas totalmente conectadas. Nessas camadas, cada neurônio está conectado a todos os neurônios da camada anterior, permitindo que a rede combine as informações extraídas e realize previsões. Essas camadas são comuns em redes voltadas para tarefas como classificação de imagens, mas em muitos casos de processamento de imagens, elas podem ser substituídas por camadas convolucionais adicionais ou redes de saída especializadas, especialmente em tarefas de segmentação ou geração de imagens. Entre as camadas convolucionais e de *pooling*, as CNNs aplicam funções de ativação como a ReLU, que é a função de ativação não-linear mais utilizada nos últimos anos, devido à sua simplicidade e ao seu comportamento semelhante ao linear, que facilita a otimização da rede (Passah et al. 2021).

As operações básicas das CNNs, como convolução e *pooling*, são matematicamente lineares. Sem uma função de ativação não-linear, a saída de cada camada seria apenas uma combinação linear das entradas, o que tornaria a rede equivalente a uma única transformação linear, incapaz de capturar relações complexas nos dados.

A ReLU ativa apenas valores positivos e zera os valores negativos, ajudando a acelerar o aprendizado e a reduzir problemas de saturação de gradiente, comuns em funções como a sigmoide. A introdução de não-linearidade é crucial para que as CNNs capturem relações não-lineares, possibilitando o aprendizado de representações hierárquicas e permitindo que a rede modele padrões complexos entre os pixels de uma imagem (Glorot et al. 2011).

2.4.2. Aplicação de CNNs no Processamento de Imagens

As CNNs têm sido amplamente utilizadas em várias tarefas de processamento de imagens, devido à sua capacidade de aprender representações discriminativas diretamente dos dados. Isso elimina a necessidade de engenheiros definirem manualmente características específicas, como bordas ou texturas, tornando-as ideais para tarefas complexas de visão computacional (Fracastoro et al. 2021).

A redução de ruído em imagens é uma tarefa clássica do processamento de imagens, e as CNNs têm oferecido avanços significativos nessa área. No caso das imagens SAR, o ruído speckle pode degradar severamente a qualidade da imagem e dificultar a

interpretação visual ou análise automática (Gonzalez & Woods 2017).

As CNNs também têm sido aplicadas com sucesso na segmentação e classificação de imagens SAR. A segmentação envolve a identificação de diferentes regiões de uma imagem, enquanto a classificação envolve a categorização dessas regiões. Redes convolucionais são capazes de aprender a reconhecer diferentes características nas imagens SAR, como áreas urbanas, corpos d'água ou vegetação, com alta precisão, mesmo na presença de ruído speckle. Há um crescente interesse em técnicas de eliminação de speckle em imagens SAR baseadas em aprendizado profundo. Métodos baseados em redes neurais convolucionais foram propostos já em 2017, e novos métodos continuam a surgir em um ritmo crescente (Cozzolino et al. 2020).

2.5. Distribuição de Probabilidade Gama e Distâncias Estocásticas

Uma distribuição de probabilidade é uma função que descreve como os valores de uma variável aleatória estão distribuídos, atribuindo uma probabilidade a cada intervalo de valores possíveis. Em termos simples, ela fornece uma representação matemática do comportamento aleatório de um sistema ou fenômeno. Distribuições de probabilidade são fundamentais na análise estatística, especialmente quando se trata de fenômenos regidos por incertezas, como o ruído presente em imagens SAR.

Como já discutido, o ruído speckle segue uma distribuição Gama, cuja função densidade de probabilidade é apresentada na Equação 2. Essa distribuição captura a variabilidade inerente ao speckle, dependendo do parâmetro L , que representa o número de *looks*.

As distâncias estocásticas são métricas que avaliam a similaridade ou dissimilaridade entre distribuições de probabilidade. Em vez de comparar diretamente valores individuais, como nas métricas tradicionais (por exemplo, a distância euclidiana), as distâncias estocásticas analisam como os dados estão distribuídos ao longo de um domínio, fornecendo uma abordagem mais robusta para dados que seguem comportamentos aleatórios, como o ruído speckle em imagens SAR (Penna & Mascarenhas 2019).

A escolha das distâncias estocásticas consideradas neste trabalho foi baseada no estudo de Penna & Mascarenhas (2019), que utilizou essas métricas para avaliar contraste

em dados com ruído speckle. Entre as distâncias selecionadas estão a divergência de Kullback-Leibler, a divergência de Jensen-Shannon, a distância de Hellinger, a distância Bhattacharyya, a divergência de Rényi, a distância triangular, a distância harmônica média e a distância aritmético-geométrica.

A principal característica das distâncias estocásticas é que elas consideram as propriedades estatísticas dos dados, como forma e dispersão, permitindo capturar diferenças entre distribuições em contextos onde os dados apresentam alta aleatoriedade. Essas métricas são amplamente empregadas para avaliar a preservação das propriedades estatísticas das imagens originais após a filtragem do ruído speckle.

As distâncias utilizadas são descritas a seguir:

2.5.1. Divergência de Kullback-Leibler (KL)

A KL mede a diferença entre duas distribuições de probabilidade P (imagem original) e Q (imagem filtrada). É definida como:

$$D_{KL}(P \parallel Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}, \quad (3)$$

onde $P(i)$ e $Q(i)$ representam as distribuições de probabilidade. Quanto menor o valor de D_{KL} , mais próxima está a distribuição da imagem filtrada em relação à original.

2.5.2. Divergência de Jensen-Shannon (JS)

A JS é uma versão simétrica e suavizada da KL, calculada como:

$$D_{JS}(P \parallel Q) = \frac{1}{2}D_{KL}(P \parallel M) + \frac{1}{2}D_{KL}(Q \parallel M), \quad (4)$$

onde $M = \frac{P+Q}{2}$ é a média das distribuições P e Q . Esta métrica varia entre 0 (distribuições idênticas) e 1 (distribuições completamente diferentes).

2.5.3. Distância de Hellinger

A distância de Hellinger mede a similaridade entre duas distribuições:

$$H(P, Q) = \frac{1}{\sqrt{2}} \sqrt{\sum_i (\sqrt{P(i)} - \sqrt{Q(i)})^2}. \quad (5)$$

Valores baixos indicam maior similaridade entre as distribuições.

2.5.4. Distância Bhattacharyya

A distância Bhattacharyya mede a sobreposição entre duas distribuições:

$$D_B(P, Q) = -\log \sum_i \sqrt{P(i)Q(i)}. \quad (6)$$

Quanto menor o valor de D_B , maior a sobreposição entre as distribuições.

2.5.5. Divergência de Rényi

A divergência de Rényi é uma generalização da KL, ajustada por um parâmetro $\alpha > 0$, que controla o peso das diferenças entre P e Q :

$$D_\alpha(P \parallel Q) = \frac{1}{\alpha - 1} \log \sum_i P(i)^\alpha Q(i)^{1-\alpha}. \quad (7)$$

O valor de α pode ser ajustado para explorar diferentes aspectos das distribuições. Quando $\alpha \rightarrow 1$, a divergência de Rényi converge para a KL.

2.5.6. Distância Triangular

A distância triangular avalia diferenças relativas entre distribuições:

$$D_T(P, Q) = \sum_i \frac{(P(i) - Q(i))^2}{P(i) + Q(i)}. \quad (8)$$

Ela enfatiza diferenças em regiões onde $P(i) + Q(i)$ é maior.

2.5.7. Distância Harmônica Média

Essa métrica considera a média harmônica dos valores das distribuições:

$$D_H(P, Q) = \sum_i \frac{2P(i)Q(i)}{P(i) + Q(i)}. \quad (9)$$

Ela mede a compatibilidade geral entre P e Q .

2.5.8. Distância Aritmético-Geométrica

A distância aritmético-geométrica compara as médias aritmética e geométrica das distribuições:

$$D_{AG}(P, Q) = \sum_i \left(\frac{P(i) + Q(i)}{2} - \sqrt{P(i)Q(i)} \right). \quad (10)$$

Valores menores indicam maior similaridade entre P e Q .

Cada uma dessas distâncias é utilizada em estudos que avaliam diferenças entre distribuições de probabilidade, como no caso do ruído speckle em imagens SAR, permitindo avaliar teoricamente a preservação das propriedades estatísticas das imagens após a aplicação de filtros.

2.6. Base de Dados BSDS500

O BSDS500 (Arbelaez et al. 2011) (*Berkeley Segmentation Dataset and Benchmark*) é um conjunto de dados amplamente utilizado na avaliação de algoritmos de segmentação de imagem e detecção de bordas. Ele contém 500 imagens naturais de alta resolução, cobrindo uma variedade de cenários e texturas, acompanhadas de anotações manuais que ajudam na avaliação e *benchmarking* de diferentes algoritmos. As imagens no BSDS500 são representativas de cenários do mundo real, tornando o conjunto de dados uma escolha valiosa para testes em métodos de processamento de imagem (Arbeláez et al. 2011).

Embora o BSDS500 tenha sido desenvolvido principalmente para segmentação de imagens, ele é uma escolha adequada para treinar e avaliar redes neurais convolucionais no contexto de remoção de ruído devido à sua diversidade de texturas e bordas. Essas características são essenciais para garantir que os modelos sejam capazes de generalizar bem em termos de redução de ruído, preservando ao mesmo tempo detalhes importantes. O uso de um conjunto de dados com alta variabilidade permite que os modelos sejam testados em diferentes condições e situações de ruído, aumentando sua robustez e eficiência (Fracastoro et al. 2021).

A abordagem de geração de speckle sintético parte de imagens ópticas, onde a quantidade de ruído pode ser considerada negligenciável. Em seguida, utiliza-se um mo-

delo de speckle, como o descrito na Equação 1, e uma caracterização estatística do speckle, como na Equação 2, para amostrar realizações do processo de speckle e sobrepor ao dado óptico limpo. Esse procedimento é um meio simples de gerar dados que se assemelham a imagens SAR com speckle e podem ser utilizados para treinar filtros de redução de ruído multiplicativo, em vez de ruído aditivo (Fracastoro et al. 2021).

2.7. Medidas de Avaliação da Qualidade de Imagens

As medidas de qualidade de imagens podem ser divididas em subjetivas e objetivas. As subjetivas dependem da avaliação humana e são geralmente mais precisas, pois em última instância o usuário final será humano. No entanto, avaliações subjetivas são custosas e difíceis de aplicar em contextos práticos. Em contraste, as métricas objetivas, que podem ser calculadas automaticamente, são mais práticas e são amplamente utilizadas em pesquisas de processamento de imagem (Santos 2017).

As métricas objetivas são necessárias para várias razões, incluindo:

- **Comparação entre Métodos:** Permitem que os pesquisadores comparem a eficácia de diferentes algoritmos em um conjunto padrão de imagens, facilitando a escolha do método mais eficaz.
- **Otimização de Algoritmos:** Proporcionam *feedback* quantitativo, que pode ser usado para ajustar e melhorar algoritmos de processamento de imagem.
- **Validação de Modelos:** As métricas ajudam a validar se um modelo de aprendizado de máquina, como uma rede neural convolucional, está efetivamente aprendendo a reduzir o ruído e preservar características importantes da imagem

Para a compreensão das métricas descritas neste capítulo, utilizaremos as seguintes definições:

- x : imagem original (sem ruído);
- z : imagem ruidosa;
- f : imagem filtrada (estimativa da imagem original);
- $x_{i,j}, z_{i,j}, f_{i,j}$: valores dos níveis de cinza na posição (i, j) das respectivas imagens;
- $\bar{x}, \bar{z}, \bar{f}$: médias dos níveis de cinza das respectivas imagens;
- $\sigma_x, \sigma_z, \sigma_f$: desvios padrão dos níveis de cinza das respectivas imagens; e
- σ_{xf} : covariância entre a imagem original e a imagem filtrada.

Consideramos que as imagens possuem tamanho $M \times N$ pixels.

2.8. Índices de Avaliação para Remoção de Speckle

Para avaliar a eficácia de filtros na remoção de ruído speckle, são utilizadas métricas específicas como o Índice de Supressão de Speckle (SSI) e o Índice de Preservação de Média e Supressão de Speckle (SMPI).

O SSI mede a eficácia do filtro na redução do ruído speckle em regiões homogêneas da imagem. A quantidade de speckle, ou Índice de Speckle (SI), é definida como a relação entre o desvio padrão (σ_f) e a média (\bar{f}) na imagem filtrada, conforme a Equação 11 (Loizou & Pattichis 2015):

$$SI = \frac{\sigma_f}{\bar{f}} \quad (11)$$

Para normalizar o SI da imagem filtrada em relação à imagem ruidosa, utiliza-se o SSI, definido pela Equação 12:

$$SSI = \frac{\sigma_f / \bar{f}}{\sigma_z / \bar{z}} \quad (12)$$

Quando $SSI < 1$, o filtro é considerado eficaz na remoção de speckle, sendo que valores menores indicam um melhor desempenho (Shamsoddini & Trinder 2010).

O SMPI, por sua vez, é descrito pela Equação 13 e avalia a remoção de speckle enquanto preserva a média da imagem. Essa métrica corrige uma limitação do SSI, que pode superestimar a média da imagem filtrada. O SMPI é calculado como:

$$SMPI = \left(\frac{\max(\bar{f}) - \min(\bar{f})}{\bar{z}} + |\bar{z} - \bar{f}| \right) \frac{\sigma_f}{\sigma_z} \quad (13)$$

Valores baixos de SMPI indicam um bom desempenho do filtro, pois o ruído é reduzido sem alterar significativamente a média da imagem (Shamsoddini & Trinder 2010).

2.9. Outros índices de qualidade

A relação sinal-ruído de pico (PSNR) mede a relação entre a potência do sinal e a potência do ruído em uma imagem. É uma métrica objetiva muito comum e útil para

avaliar a qualidade de imagens filtradas, sendo expressa pela equação 14:

$$PSNR = 10 \log_{10} \left(\frac{g_{\max}^2}{MSE} \right) \quad (14)$$

onde g_{\max} é o valor máximo possível de pixel (por exemplo, 255 para imagens em tons de cinza de 8 bits). O Erro Quadrático Médio (MSE) é dado pela equação 15:

$$MSE = \frac{1}{M \times N} \sum_{i,j} (z_{i,j} - f_{i,j})^2 \quad (15)$$

em que o termo $M \times N$ representa o **número total de pixels na imagem**, onde:

- M é o número de linhas (altura da imagem);
- N é o número de colunas (largura da imagem).

Um valor elevado de PSNR indica uma maior proporção de sinal em relação ao ruído, refletindo uma qualidade superior na imagem filtrada (Santos 2017).

O Índice de Similaridade Estrutural (SSIM) mede a qualidade da imagem considerando aspectos perceptuais. Inspirado pelo sistema visual humano, o SSIM avalia similaridades de luminância, contraste e estrutura entre duas imagens. O SSIM é representado pela equação 16 abaixo:

$$SSIM = \frac{(2\bar{x}\bar{f} + C_1)(2\sigma_{xf} + C_2)}{(\bar{x}^2 + \bar{f}^2 + C_1)(\sigma_x^2 + \sigma_f^2 + C_2)} \quad (16)$$

onde $C_1 = (0.01 \times r)^2$ e $C_2 = (0.03 \times r)^2$, com r sendo o intervalo dinâmico da imagem (para tons de cinza em 8 bits, $r = 255$). O SSIM varia de -1 (totalmente dissimilares) a 1 (totalmente similares), sendo que valores mais altos indicam uma filtragem eficaz que preserva a estrutura visual (Santos 2017).

3. Revisão de literatura

Este capítulo propõe uma revisão de literatura sobre o uso de redes neurais para filtrar o ruído speckle em imagens SAR. Para garantir uma análise abrangente e sistemática dos estudos existentes, adotamos uma abordagem baseada nos princípios da revisão sistemática. Seguindo os conceitos apresentados por Kitchenham & Charters (2007), esta revisão emprega critérios pré-definidos e explícitos para a seleção de artigos relevantes da literatura científica.

O processo de revisão sistemática inclui diversas etapas, desde a formulação da questão de pesquisa até a síntese e avaliação dos resultados obtidos. Neste contexto, exploramos estudos relacionados ao tema, escolhidos criteriosamente com base em critérios de inclusão e exclusão bem definidos. Ao final, esta revisão visa oferecer uma visão abrangente e confiável do estado atual da pesquisa sobre o uso de redes neurais para filtrar o ruído speckle em imagens SAR.

3.1. Planejamento

O planejamento detalhado de uma revisão de literatura é fundamental para garantir a precisão e a eficácia do processo (Biolchini et al. 2005). Nesta seção, descrevemos as estratégias adotadas para conduzir a revisão sistemática sobre o uso de redes neurais para filtrar ruído speckle em imagens SAR.

Na etapa de planejamento foi decidido que só seriam considerados os trabalhos publicados nos últimos 5 anos nos repositórios Google Scholar, IEEE, ACM e Science Direct.

As questões norteadoras foram formuladas com o objetivo de orientar a revisão e direcionar a análise dos estudos selecionados. As questões norteadoras são:

- Quais são as abordagens mais recentes baseadas em redes neurais para filtragem de ruído speckle em imagens de radar de abertura sintética (SAR)?
- Como as redes neurais convolucionais (CNNs) têm sido aplicadas na redução de ruído speckle em imagens SAR?

Para identificar estudos relevantes, utilizamos uma *string* de busca cuidadosamente elaborada, que inclui os termos-chave relacionados ao tema de interesse. A *string*

de busca adotada foi:

("Neural Network") AND ("Despeckling"OR "Image Denoising") AND ("Synthetic Aperture Radar Images"OR "SAR")

Essa *string* de busca retornou uma quantidade satisfatória de resultados, trazendo uma pesquisa relevante para este trabalho.

Os critérios de inclusão foram estabelecidos para garantir a seleção de estudos relevantes e de alta qualidade para análise. Os critérios de inclusão adotados são apresentados na Tabela 1.

Tabela 1. Critérios de Inclusão

Critérios de Inclusão	Descrição
Abordagem Baseada em Redes Neurais	Incluir apenas estudos que se concentram principalmente em abordagens baseadas em redes neurais para filtragem de ruído speckle em imagens SAR.
Avaliação Quantitativa e Qualitativa	Incluir estudos que fornecem uma avaliação clara da eficácia das técnicas propostas, seja por meio de métricas quantitativas ou avaliação qualitativa.
Data de Publicação	Incluir estudos recentes (últimos 5 anos) para incorporar as mais recentes inovações e avanços na área.
Língua	Incluir apenas estudos disponíveis em línguas que possam ser compreendidas ou traduzidas facilmente para facilitar a revisão e análise.

Os critérios de exclusão foram definidos para filtrar estudos que não atendem aos requisitos ou que apresentam limitações específicas. Os critérios de exclusão adotados são apresentados na Tabela 2.

3.2. Execução da Revisão

O processo de execução da revisão foi conduzido em quatro etapas distintas, conforme descrito a seguir.

3.2.1. Busca e Seleção de Artigos

Inicialmente, foram realizadas buscas nos repositórios acadêmicos usando a *string* de busca definida durante o planejamento. Os resultados foram obtidos em quatro repositórios: Google Scholar, IEEE Xplore, ACM Digital Library e Science Direct. O número total de artigos obtidos em cada repositório nesta etapa é apresentado na Tabela 3.

Tabela 2. Critérios de Exclusão

Critérios de Exclusão	Descrição
Abordagens Não Baseadas em Redes Neurais	Excluir estudos que se concentram exclusivamente em métodos que não envolvem redes neurais, a menos que sejam discutidos em comparação com abordagens baseadas em redes neurais.
Foco em Outros Tipos de Ruído	Excluir estudos que tratam predominantemente de técnicas de remoção de ruído que não estejam relacionadas ao ruído speckle em imagens SAR.
Estudos sem Avaliação de Desempenho	Excluir trabalhos que não apresentem uma avaliação clara do desempenho das abordagens propostas.
Aplicações Não Relacionadas	Excluir estudos cujas abordagens não estejam diretamente relacionadas a aplicações de imagens SAR ou à filtragem de ruído speckle.
Relatos de Casos e Resumos de Conferências	Excluir relatos de casos, resumos de conferências e outros formatos que não forneçam informações suficientes para uma análise aprofundada.
Publicações Antigas e Desatualizadas	Excluir estudos publicados antes do período especificado (últimos 5 anos) e aqueles que não apresentem relevância atual para a pesquisa.

Tabela 3. Resultados da Busca e Seleção de Artigos

Repositório	Artigos Obtidos
Google Scholar	4280
IEEE Xplore	113
ACM Digital Library	41
Science Direct	177

3.2.2. Filtragem por Relevância e Aplicação de Critérios

Após a obtenção dos resultados da busca, foram aplicados os critérios de inclusão e exclusão definidos durante o planejamento para selecionar os artigos mais relevantes. Como a busca gerou um número elevado de resultados em alguns repositórios, foi decidido focar na análise dos 50 primeiros artigos em cada repositório.

Esse número foi escolhido por ser representativo e manejável, garantindo que os artigos mais relevantes e ordenados por ferramentas de busca, como relevância e número de citações, fossem priorizados. Essa abordagem permitiu um equilíbrio entre abrangência e viabilidade no processo de análise.

Os resultados desta etapa, após a aplicação dos critérios, são apresentados na Ta-

bela 4.

Tabela 4. Resultados após Filtragem por Relevância e Aplicação de Critérios

Repositório	Artigos Selecionados
Google Scholar	24
IEEE Xplore	50
ACM Digital Library	10
Science Direct	50

3.2.3. Análise dos Artigos Selecionados

Os artigos selecionados na etapa anterior foram submetidos a uma análise mais detalhada, considerando seus títulos, resumos e palavras-chave em relação às questões norteadoras definidas. Esta análise resultou na seleção de um número reduzido de artigos para a próxima etapa, conforme mostrado na Tabela 5.

Tabela 5. Resultados após Análise dos Artigos Selecionados

Repositório	Artigos Selecionados
Google Scholar	15
IEEE Xplore	40
ACM Digital Library	2
Science Direct	12

3.2.4. Avaliação Final e Seleção dos Artigos Definitivos

Os artigos selecionados na etapa anterior foram analisados com maiores detalhes e submetidos novamente aos critérios de inclusão e exclusão. Após essa avaliação final, foram selecionados os artigos definitivos que foram incluídos na revisão. O número final de artigos selecionados em cada repositório é apresentado na Tabela 6.

Tabela 6. Resultados Finais após Avaliação e Seleção dos Artigos Definitivos

Repositório	Artigos Selecionados
Google Scholar	9
IEEE Xplore	16
ACM Digital Library	1
Science Direct	2

3.3. Resultados da revisão

Os artigos selecionados oferecem uma visão abrangente e detalhada dos métodos mais recentes e inovadores para o *despeckling* de imagens SAR usando redes neurais convolucionais. Cada abordagem tem suas próprias características e vantagens, desde a combinação de diferentes arquiteturas de CNN até o uso de estratégias de aprendizado auto-supervisionado.

Alguns artigos se concentram em melhorar a eficácia do *despeckling* em condições de ruído intenso, como aqueles que propõem métodos para *despeckling* simultâneo e classificação de alvos SAR. Outros exploram técnicas avançadas, como aprendizado residual, redes residuais de atenção e codificação paralela, para alcançar resultados superiores em imagens SAR sintéticas e reais.

Além disso, há estudos que abordam desafios específicos, como a construção de conjuntos de dados de treinamento para aprendizado profundo em SAR, o uso de modelos pré-treinados e estratégias de treinamento híbrido para melhorar a generalização e eficácia dos algoritmos de *despeckling*.

Esses avanços demonstram o potencial das CNNs para resolver problemas complexos de processamento de imagens SAR e pavimentar o caminho para aplicações mais precisas e eficientes nessa área.

Foram destacados 5 artigos que estão dispostos na Tabela 7 que destacam diferentes abordagens para a remoção de ruído speckle em imagens SAR utilizando redes neurais convolucionais. Abaixo estão descrições resumidas de cada estudo:

1. Dalsasso et al. (2020) destacam a eficácia de métodos baseados em CNNs. O artigo propõe estratégias que incluem o uso de modelos pré-treinados e abordagens híbridas para melhorar a remoção de speckle. Os resultados são comparados com filtros do estado-da-arte.
2. Liu & Liu (2022) propõem o algoritmo M-CNN, que combina convoluções dilatadas, ativação LReLU e aprendizado residual. O método incorpora conexões skip para evitar degradações na qualidade da imagem.
3. Passah et al. (2021) apresentam um modelo de *despeckling* baseado em redes neurais residuais com uma função de perda híbrida. Os resultados experimentais mos-

tram ganhos significativos em PSNR em diferentes condições de observação.

4. Vitale et al. (2021) introduzem uma função de custo multiobjetivo em uma CNN, combinando propriedades espaciais e estatísticas para melhorar a precisão da filtragem.
5. Cozzolino et al. (2020) propõem um filtro não local combinado com aprendizado profundo, utilizando uma CNN para atribuir pesos adaptativos, apresentando resultados superiores em imagens simuladas e reais.

Tabela 7. Artigos Selecionados sobre Filtragem de Speckle em Imagens SAR

Referência	Título
(Dalsasso et al. 2020)	SAR Image Despeckling by Deep Neural Networks: from a Pre-Trained Model to an End-to-End Training Strategy
(Liu & Liu 2022)	Synthetic aperture radar image despeckling using convolutional neural networks in wavelet domain
(Passah et al. 2021)	SAR image despeckling using deep CNN
(Vitale et al. 2021)	Multi-Objective CNN-Based Algorithm for SAR Despeckling
(Cozzolino et al. 2020)	Nonlocal CNN SAR Image Despeckling

3.4. Conclusão da Revisão

Esta revisão sistemática identificou e analisou as abordagens mais recentes baseadas em redes neurais para a filtragem de ruído speckle em imagens SAR. O processo estruturado de busca, seleção e análise permitiu avaliar métodos relevantes e alinhados às questões de pesquisa.

As CNNs mostraram-se eficazes na redução de ruído speckle, preservando detalhes estruturais das imagens SAR. Arquiteturas avançadas e técnicas híbridas foram identificadas como áreas em crescimento, com resultados superiores em métricas como PSNR e SSIM, evidenciando o avanço tecnológico nesse domínio.

Além disso, a revisão apontou desafios a serem enfrentados, como a necessidade de conjuntos de dados mais amplos e representativos, bem como o desenvolvimento de métodos que equilibrem a eficácia na remoção de ruído com a preservação da integridade das características da imagem.

4. Metodologia

A metodologia desenvolvida neste trabalho consiste em um pipeline completo para a filtragem de ruído speckle em imagens SAR, combinando técnicas de aprendizado profundo e avaliação com distâncias estocásticas. As etapas principais incluem preparação de dados, simulação de ruído, treinamento e validação de um modelo baseado em redes neurais convolucionais, e avaliação quantitativa e qualitativa dos resultados.

1.Preparação dos Dados:

Utilização do banco de dados BSDS500 para criar um conjunto de imagens contendo ruído tipo speckle. O processo inclui pré-processamento, adição de ruído artificial e divisão em conjuntos de treinamento e validação.

2.Construção e Treinamento do Modelo:

Implementação de uma rede neural convolucional em Python, utilizando o *framework* PyTorch (Paszke et al. 2019). O modelo é treinado para remover o ruído speckle das imagens ruidosas, aprendendo a recuperar as características originais das imagens.

3.Análise das Distâncias Estocásticas:

Cálculo de distâncias estocásticas entre as distribuições de intensidade das imagens filtradas e das imagens originais, como forma de avaliar a preservação das propriedades estatísticas das imagens durante a filtragem.

4.Avaliação de Desempenho:

Validação do modelo utilizando métricas tradicionais de qualidade de imagem, como PSNR e SSIM, e análise qualitativa dos resultados visuais.

O processo combina técnicas de aprendizado profundo com distâncias estocásticas para analisar e comparar a eficácia da remoção de ruído, utilizando índices de qualidade para validação.

4.1. Preparação de Dados

A preparação de dados é uma etapa essencial para o desenvolvimento do modelo, pois garante a consistência e a qualidade das informações fornecidas à rede neural. Esta seção descreve os passos realizados para carregar, processar e organizar as imagens utilizadas no treinamento e validação do modelo.

4.1.1. Conjunto de Dados: BSDS500

O conjunto de dados utilizado neste trabalho é o BSDS500, um banco amplamente reconhecido por sua diversidade de imagens naturais de alta resolução. As imagens foram extraídas da pasta de treinamento do BSDS500 e convertidas para escala de cinza, considerando que o objetivo deste trabalho é focar em imagens monocromáticas, como as de radar SAR.

4.1.2. Redimensionamento das Imagens

As imagens originais possuem dimensões variadas, o que poderia dificultar o processamento pela rede neural. Para padronizar os dados e facilitar o treinamento, todas as imagens foram redimensionadas para 128×128 pixels. Essa resolução foi escolhida como um equilíbrio entre a preservação dos detalhes das imagens e a redução do custo computacional.

4.1.3. Adição de Ruído Speckle

Para simular a degradação causada pelo ruído speckle, as imagens limpas foram artificialmente contaminadas utilizando um modelo multiplicativo. Nesse modelo, a intensidade observada em cada pixel é calculada como o produto entre a intensidade original e um fator aleatório gerado por uma distribuição estatística gamma, garantindo a reprodução fiel das características do ruído speckle.

O ruído foi inserido como uma variação aleatória nos valores de intensidade dos pixels, reproduzindo padrões de ruído específicos. Para essa tarefa, foi utilizado o módulo `random` da biblioteca NumPy, que é uma ferramenta amplamente adotada para a geração de números aleatórios e simulação de processos estocásticos (Sarkar & Sudeep 2024).

No contexto deste trabalho, o número de *looks* L desempenha um papel fundamental na definição da intensidade do ruído speckle. Uma baixa variância gera ruído leve, resultando em pequenas flutuações nos valores de intensidade dos pixels, enquanto uma alta variância provoca um ruído mais severo, causando variações significativas. Dessa forma, a variância controla diretamente a severidade do ruído introduzido, permitindo ajustar o nível de degradação nas imagens para fins de treinamento e avaliação do modelo.

4.1.4. Normalização das Imagens

Após a adição do ruído, os valores dos pixels das imagens foram normalizados para o intervalo $[0,1]$. Essa normalização é uma prática comum em aprendizado de máquina, pois ajuda a estabilizar o processo de treinamento, reduzindo a sensibilidade do modelo a diferenças na escala dos valores.

4.1.5. Divisão em Conjuntos de Treinamento e Validação

O conjunto de dados foi dividido em dois subconjuntos:

- Treinamento:** Usado para ajustar os parâmetros do modelo durante o treinamento.
- Validação:** Usado para avaliar o desempenho do modelo em dados não vistos durante o treinamento, prevenindo problemas como *overfitting*.

A divisão foi realizada de forma aleatória, mantendo a distribuição equilibrada das características das imagens entre os conjuntos.

Esse processo garantiu uma amostra suficiente para o treinamento e avaliação do modelo, respeitando as limitações impostas pelo tamanho reduzido do *dataset* utilizado.

4.1.6. Organização dos Dados para Treinamento

Para fornecer os dados de forma eficiente à rede neural, foi criado um *dataset* customizado que encapsula as imagens ruidosas (entrada do modelo) e suas respectivas versões limpas (saída esperada). Este *dataset*, implementado com base na classe `Dataset` do PyTorch, permite acessar pares de imagens de forma indexada e organizada.

O *DataLoader* (PyTorch Contributors 2023) do PyTorch foi utilizado para gerenciar o fluxo de dados durante o treinamento e a validação. Ele foi configurado com os seguintes parâmetros:

- Mini-batches:** Cada lote contém 4 pares de imagens ruidosas e sem ruído.
- Embaralhamento:** No treinamento, os dados foram embaralhados a cada época (`shuffle=True`) para evitar que o modelo aprenda padrões relacionados à ordem das imagens.

- **Validação ordenada:** Os dados de validação foram processados na ordem original (`shuffle=False`), garantindo consistência na avaliação do modelo.

Essa abordagem permite um processamento eficiente dos dados, reduzindo o custo computacional por iteração e maximizando a utilização dos recursos de hardware, como a GPU.

4.2. Desenvolvimento e Treinamento do Modelo

Nesta etapa, descrevemos o processo de construção e treinamento do modelo de rede neural convolucional para remover o ruído speckle de imagens contaminadas. Este modelo foi projetado para aprender a mapear imagens ruidosas em suas versões limpas, utilizando aprendizado supervisionado.

4.2.1. Desenvolvimento do Modelo

Arquitetura da Rede Neural O modelo implementado é uma rede neural convolucional customizada, projetada especificamente para tarefas de filtragem de ruído em imagens. A arquitetura da rede inclui:

- **Camada de Entrada:** Uma convolução inicial com 32 filtros 3×3 , responsável por capturar as características básicas da imagem, seguida de uma função de ativação ReLU para introduzir não linearidade.
- **Camadas Intermediárias:** Sete camadas convolucionais subsequentes, cada uma com 32 filtros 3×3 , normalização em lote (*Batch Normalization*) para estabilizar o treinamento, e função de ativação ReLU. Essas camadas são responsáveis por aprender padrões mais complexos, como bordas e texturas.
- **Camada de Saída:** Uma camada convolucional final com um único filtro 3×3 , que reconstrói a imagem filtrada no mesmo tamanho e escala da entrada.

A arquitetura foi projetada para um equilíbrio entre profundidade suficiente para capturar padrões complexos e simplicidade para evitar *overfitting*.

4.2.2. Treinamento do Modelo

O treinamento do modelo foi realizado utilizando o algoritmo de otimização Adaptive Moment Estimation (Adam), reconhecido por sua eficiência em cenários de aprendizado profundo (Kingma & Ba 2015).

O Adam (*Adaptive Moment Estimation*) é um algoritmo de otimização amplamente utilizado em aprendizado profundo devido à sua eficiência e robustez. Ele combina as vantagens do *Momentum*, que utiliza a média dos gradientes passados para acelerar a convergência, e da adaptação da taxa de aprendizado, que ajusta automaticamente a taxa de aprendizado para cada parâmetro com base na variância recente dos gradientes.

Especificamente, o Adam calcula estimativas em tempo real da média (para rastrear a direção geral dos gradientes) e da variância (para medir sua magnitude). Essas estimativas permitem atualizações dinâmicas e estáveis dos pesos durante o treinamento, reduzindo a necessidade de ajustes manuais de hiperparâmetros, como a taxa de aprendizado inicial. Essa característica o torna uma escolha padrão em muitas aplicações de aprendizado profundo (Kingma & Ba 2015).

Esse processo foi repetido para cada mini-batch de imagens e para cada época, com o objetivo de reduzir progressivamente o valor da função de perda combinada. A função de perda incluiu o erro quadrático médio e uma composição de métricas estocásticas:

- Divergência de Kullback-Leibler, para medir a discrepância entre distribuições originais e filtradas.
- Distância de Hellinger, para avaliar similaridades geométricas.
- Divergência de Jensen-Shannon, uma métrica suavizada e simétrica.

Cada métrica foi ponderada na função de perda para refletir diferentes aspectos da qualidade das imagens filtradas. O modelo foi treinado em um total de 10 épocas, com lotes de 4 imagens por iteração.

Durante a etapa de validação, o modelo foi configurado em *modo de avaliação*, o que desativou operações específicas do treinamento, como a normalização em lote (*batch normalization*) e o *dropout*. Essa configuração garante que a avaliação reflita de forma precisa o comportamento esperado do modelo em um ambiente de produção, onde essas operações geralmente não estão ativas.

A perda média no conjunto de validação foi calculada e comparada à perda média do conjunto de treinamento, permitindo avaliar a capacidade do modelo de generalizar para dados não vistos. Esse processo é essencial para identificar discrepâncias que possam

indicar problemas na configuração do treinamento.

Essa análise permitiu monitorar o comportamento do modelo em dados não vistos e identificar possíveis sinais de *overfitting*, como uma perda de validação significativamente maior que a de treinamento. Caso fosse detectado, ajustes nos hiperparâmetros, como a taxa de aprendizado, número de épocas ou o uso de regularização, poderiam ser realizados.

4.3. Avaliação com Distâncias Estocásticas

Além das métricas tradicionais, distâncias estocásticas foram calculadas para avaliar a preservação das propriedades estatísticas das imagens. Estas métricas medem a similaridade entre as distribuições de intensidade das imagens originais e filtradas.

4.4. Análise Qualitativa e Quantitativa

A avaliação foi conduzida em duas vertentes: quantitativa e qualitativa. Métricas como PSNR e SSIM foram utilizadas para avaliar a preservação de detalhes e a qualidade da filtragem. Além disso, distâncias estocásticas forneceram uma análise aprofundada sobre a similaridade entre as distribuições de pixel das imagens originais e filtradas.

A análise qualitativa foi realizada por meio da visualização das imagens originais, ruidosas e filtradas, permitindo verificar aspectos como:

- Preservação de bordas:** A capacidade do modelo de manter detalhes estruturais, como linhas e contornos.
- Redução do ruído:** A eficácia na atenuação do ruído speckle sem criar artefatos visuais.
- Naturalidade da imagem:** Garantir que a imagem filtrada não perca texturas ou pareça artificial.

Essas imagens foram exibidas lado a lado para facilitar a comparação.

4.5. Implementação do Código

Para implementar o modelo de rede neural convolucional descrito neste projeto, foi necessário construir um *script* em Python, utilizando bibliotecas especializadas para processamento de imagens, aprendizado de máquina e manipulação de dados (Dhruv

et al. 2020). Este capítulo detalha os componentes fundamentais do código, incluindo as bibliotecas utilizadas e as funções implementadas.

4.5.1. Bibliotecas Utilizadas no Projeto

A implementação deste projeto requer o uso de várias bibliotecas Python, cada uma com um propósito específico e essencial para o funcionamento do modelo. A seguir, detalhamos as bibliotecas utilizadas:

- 1.NumPy: é uma biblioteca fundamental para computação científica em Python, amplamente utilizada para manipulação de *arrays* multidimensionais e realização de operações matemáticas eficientes. Neste trabalho, foi empregada para operações estatísticas, geração de ruído speckle e normalização das imagens;
- 2.OpenCV: é uma biblioteca poderosa para processamento de imagens e visão computacional. Foi utilizada para tarefas como leitura, redimensionamento e conversão das imagens para escala de cinza, garantindo que todas as imagens tivessem o formato e as dimensões apropriadas para o treinamento do modelo;
- 3.PyTorch: é uma biblioteca de aprendizado profundo que oferece suporte a computação acelerada por GPU e flexibilidade para criar e treinar redes neurais. Foi utilizada para definir a arquitetura do modelo, treinar a rede neural e manipular tensores;
- 4.scikit-learn: é uma ferramenta versátil para aprendizado de máquina e análise de dados. Foi utilizada para dividir o conjunto de dados em treinamento e validação de forma eficiente e reproduzível;
- 5.SciPy: oferece funcionalidades avançadas para computação científica e estatística. Foi empregada para calcular métricas estocásticas, como a divergência de Kullback-Leibler e a distância de Jensen-Shannon, permitindo uma análise detalhada das distribuições de intensidade das imagens;
- 6.Matplotlib: é uma biblioteca para visualização de dados, usada para exibir os resultados do modelo. As imagens originais, ruidosas e filtradas foram apresentadas lado a lado para facilitar a análise qualitativa e a comparação visual;
- 7.tqdm: é uma biblioteca utilizada para criar barras de progresso em loops longos. Foi integrada ao processo de treinamento para monitorar visualmente o progresso de cada época e fornecer *feedback* em tempo real;

- 8.scikit-image: é uma biblioteca específica para processamento de imagens, que oferece uma ampla gama de funções para análise, transformação e manipulação de imagens;
- 9.torch.cuda.amp: Fornece suporte para treinamento com precisão mista, reduzindo o consumo de memória e acelerando o treinamento ao equilibrar cálculos em precisão simples e dupla.

4.6. Apresentação do Script

O script conta com uma série de funções principais que estruturam o pipeline para a filtragem de ruído speckle em imagens SAR. Essas funções foram organizadas para garantir modularidade e facilitar a reprodutibilidade do método. No Apêndice A, está apresentado os detalhes do script desenvolvido, incluindo explicações sobre as bibliotecas utilizadas, exemplos de código e resultados obtidos.

5. Experimentos, Resultados e Discussões para Validação

Este capítulo apresenta a execução prática do modelo proposto para a remoção de ruído speckle de imagens SAR, bem como os métodos utilizados para avaliação de seu desempenho. O foco está na aplicação dos processos planejados para posterior análise dos resultados obtidos.

5.1. Preparação dos Dados

As imagens do conjunto BSDS500 foram redimensionadas para 128×128 pixels e normalizadas no intervalo $[0, 1]$ para equilibrar a preservação dos detalhes visuais com a eficiência computacional. As imagens contidas no conjunto foram usadas para acelerar a validação do modelo usado e pela facilidade de obter-las e trata-las. O ruído speckle foi artificialmente adicionado utilizando um modelo multiplicativo, simulando de maneira realista o ruído presente em imagens SAR.

Os DataLoaders foram utilizados para organizar os dados em mini-batches de 4 imagens, otimizando o uso de recursos computacionais e facilitando o treinamento em um ambiente de hardware limitado.

5.2. Treinamento do Modelo

O treinamento foi conduzido utilizando uma rede neural convolucional projetada para realizar a filtragem adaptativa do ruído. O otimizador Adam foi empregado para ajustar os pesos da rede com base na minimização do Erro Quadrático Médio, que mede a diferença entre as imagens filtradas geradas pelo modelo e as imagens limpas.

O MSE foi escolhido por ser uma métrica amplamente utilizada em problemas de regressão, como a filtragem de ruído, pois penaliza de forma mais severa os erros maiores (diferenças mais significativas entre os valores previstos e os reais). Isso é especialmente relevante no contexto da remoção de ruído speckle, onde preservar a precisão dos detalhes estruturais e minimizar grandes desvios nos valores de intensidade são objetivos prioritários.

O processo foi repetido por 10 épocas, monitorando a perda em conjuntos de treinamento e validação para avaliar a capacidade de generalização do modelo.

O sistema utilizado para o treinamento incluiu um sistema operacional Windows 10, com 8 GB de RAM e suporte a GPU dedicada. A implementação foi feita utilizando a biblioteca PyTorch e o ambiente de desenvolvimento Visual Studio Code.

5.3. Avaliação do Modelo

A avaliação do modelo foi realizada de maneira quantitativa e qualitativa. Para análise quantitativa, foram calculados os índices PSNR e SSIM, que medem, respectivamente, a relação sinal-ruído de pico e a preservação de características estruturais das imagens. Além disso, distâncias estocásticas foram calculadas, fornecendo informações adicionais sobre a similaridade estatística entre as distribuições das intensidades das imagens originais e filtradas.

Na análise qualitativa, as imagens originais, ruidosas e filtradas foram exibidas lado a lado. Essas visualizações permite avaliar a capacidade do modelo de reduzir o ruído speckle enquanto preservava bordas e texturas relevantes.

5.4. Configuração Experimental

Os experimentos foram conduzidos utilizando o modelo desenvolvido, implementado em Python com suporte de bibliotecas amplamente reconhecidas no campo do aprendizado profundo.

Os experimentos realizados buscaram validar a metodologia proposta em um ambiente computacional acessível, utilizando um computador de uso pessoal com hardwares comuns, sem recursos especializados. Isso demonstra a viabilidade de implementação do modelo em equipamentos amplamente disponíveis, reforçando a acessibilidade da solução.

5.4.1. Configurações de Hardware e Software

As especificações do ambiente experimental estão apresentadas na Tabela 8:

Apesar das limitações de desempenho inerentes a um computador de uso pessoal, as configurações mostraram-se suficientes para executar o treinamento e a avaliação do modelo, ainda que o tempo de processamento tenha sido superior ao que seria alcançado em hardwares especializados, como GPUs de alto desempenho.

Tabela 8. Configurações de Hardware e Software

Categoria	Especificação
Hardware	
Processador	Intel Core i7
GPU	AMD Radeon HD 7400M Série
Memória RAM	8 GB
Software	
Sistema Operacional	Windows 10
Python	Versão 3.9
Bibliotecas	NumPy, OpenCV, PyTorch, scikit-learn, SciPy, Matplotlib, tqdm

5.4.2. Configurações de Treinamento

Os parâmetros utilizados no treinamento do modelo foram definidos para alcançar um equilíbrio entre eficiência e custo computacional. A Tabela 9 apresenta os principais aspectos do treinamento:

Tabela 9. Configurações de Treinamento

Parâmetro	Descrição
Conjunto de Dados	BSDS500, com 90% para treinamento e 10% para validação.
Arquitetura da Rede Neural	Rede convolucional com camadas convolucionais, normalização em lote e ativação ReLU.
Função de Perda	Combinação do Erro Quadrático Médio e métricas estocásticas.
Otimizador	Adam, com taxa de aprendizado de 0.001.
Número de Épocas	10
Tamanho do Lote	4 imagens por iteração.

Com essas configurações, o modelo foi treinado e avaliado, assegurando a reprodutibilidade dos experimentos e permitindo uma análise detalhada dos resultados obtidos.

5.5. Resultados Quantitativos

Os resultados quantitativos dos experimentos foram obtidos utilizando as métricas PSNR e SSIM, amplamente reconhecidas para avaliar a qualidade de imagens processadas.

Para avaliar o desempenho do modelo, foi realizada uma combinação de testes variando três parâmetros principais: o número de *looks*, a quantidade de imagens uti-

lizadas no treinamento e o número de épocas de treinamento. Foram testadas todas as combinações possíveis dos seguintes valores:

- Número de *looks*: 1 e 5;
- Quantidade de imagens de treinamento: 20, 100 e 200;
- Número de épocas: 1, 10 e 100.

No total, foram realizadas 18 combinações, sendo que cada configuração foi repetida 10 vezes para garantir resultados mais precisos e estatisticamente confiáveis. No entanto, para as combinações que envolveram 100 épocas e 200 imagens, os experimentos foram realizados apenas uma vez devido ao alto tempo computacional necessário para o treinamento nessas condições.

A seguir, as tabelas apresentam os valores médios de PSNR e SSIM para cada combinação de parâmetros.

Tabela 10. Resultados quantitativos para *Looks* = 1

Imagens	Épocas	PSNR (dB)	SSIM
20	1	13.4227	0.2854
20	10	17.2456	0.3938
20	100	18.3210	0.4213
100	1	14.5210	0.3156
100	10	19.8450	0.4551
100	100	20.1123	0.4705
200	1	15.2234	0.3522
200	10	20.4567	0.4823
200	100	20.3124	0.4819

Tabela 11. Resultados quantitativos para *Looks* = 5

Imagens	Épocas	PSNR (dB)	SSIM
20	1	15.4227	0.3274
20	10	18.2183	0.4385
20	100	19.3121	0.4671
100	1	16.8123	0.3750
100	10	20.9234	0.4903
100	100	21.0156	0.4932
200	1	17.5891	0.4055
200	10	21.7234	0.5111
200	100	21.8123	0.5105

Os resultados mostram que:

- Valores de PSNR e SSIM são maiores para imagens com 5 *looks* em comparação com as equivalentes de 1 *look*, indicando que o modelo tem melhor desempenho em imagens menos impactadas pelo ruído speckle.
- O desempenho melhora consistentemente à medida que aumenta o número de imagens utilizadas para treinamento e o número de épocas de treinamento, especialmente para configurações com poucos dados iniciais.
- Em configurações com 100 épocas e muitas imagens (100 ou 200), os resultados de PSNR e SSIM tendem a estabilizar. Isso reflete o fenômeno de *overfitting*, em que o modelo se adapta excessivamente aos dados de treinamento e apresenta ganhos marginais em dados de teste. Esse comportamento sugere que um limite ótimo de treinamento deve ser buscado para evitar desperdício de recursos computacionais.

Esses resultados destacam o impacto de cada parâmetro no desempenho do modelo, sendo a filtragem de imagens *single-look* o principal desafio devido ao alto nível de ruído speckle.

5.6. Resultados Qualitativos

A análise qualitativa dos resultados fornece uma visão complementar às métricas quantitativas, permitindo avaliar visualmente o desempenho do modelo na redução do ruído speckle e na preservação das características estruturais das imagens SAR.

5.6.1. Imagem Original

A imagem original, apresentada na Figura 5, é utilizada como referência para comparação com as imagens ruidosas e filtradas. Essa imagem não contém ruído speckle e preserva bordas, texturas e características estruturais detalhadas, servindo como base para avaliar o impacto do modelo e do ruído em diferentes condições.



Figura 5. Imagem sem ruído e normalizada, utilizada como referência.

5.6.2. Imagens Contaminadas com Ruído Speckle

Duas versões da imagem original foram contaminadas artificialmente com ruído speckle. A Figura 6 apresenta, à esquerda, a imagem contaminada com $L = 1$ (*single-look*), onde o ruído speckle é altamente granular, tornando a imagem visualmente mais degradada e desafiadora para o processo de filtragem. Já à direita, a mesma imagem foi contaminada com $L = 5$ (*multi-look*), resultando em um ruído significativamente menos perceptível, o que reflete condições mais favoráveis para a filtragem.

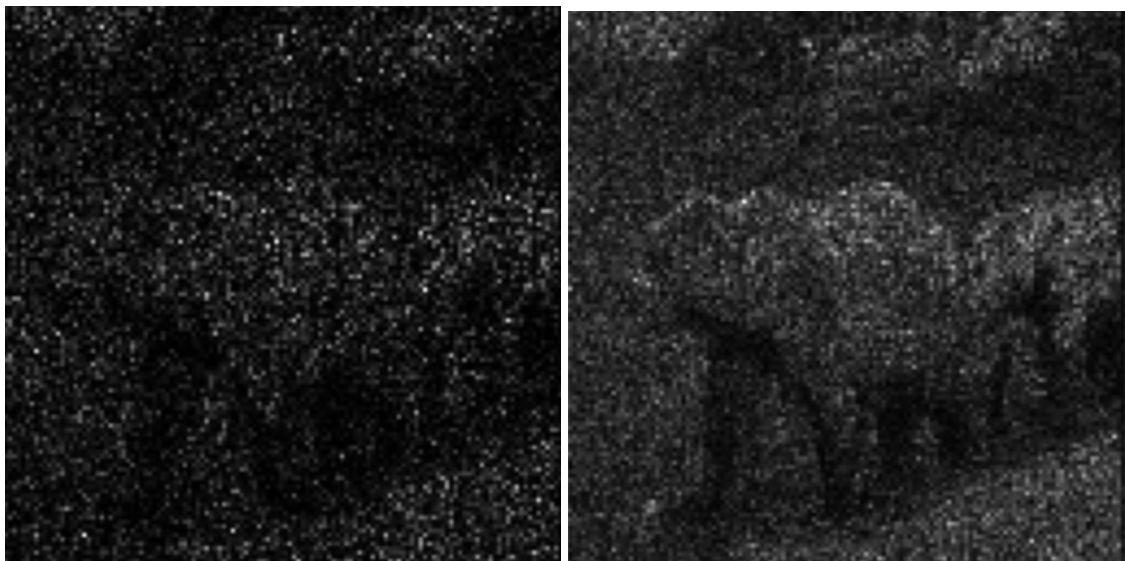


Figura 6. À esquerda: Imagem contaminada com ruído speckle, $L = 1$. À direita: Imagem contaminada com ruído speckle, $L = 5$.

5.6.3. Resultados de Filtragem

Por fim, a Figura 7 apresenta três exemplos de filtragem realizados pelo modelo, representando diferentes condições de treinamento e teste. À esquerda, observa-se o resultado para $L = 1$, com 20 imagens de treinamento e apenas 1 época. Nesse caso, o ruído residual ainda é evidente, e a preservação de bordas e texturas é limitada, caracterizando um desempenho insatisfatório.

No exemplo central, correspondente a $L = 5$, com 100 imagens de treinamento e 10 épocas, o modelo foi capaz de reduzir o ruído de forma moderada. No entanto, ainda há perda parcial de detalhes em regiões de alto contraste.

Por fim, à direita, está o resultado para $L = 5$, com 200 imagens de treinamento e 100 épocas. Aqui, o modelo conseguiu alcançar uma filtragem mais eficiente, preservando bordas e texturas com melhor qualidade.

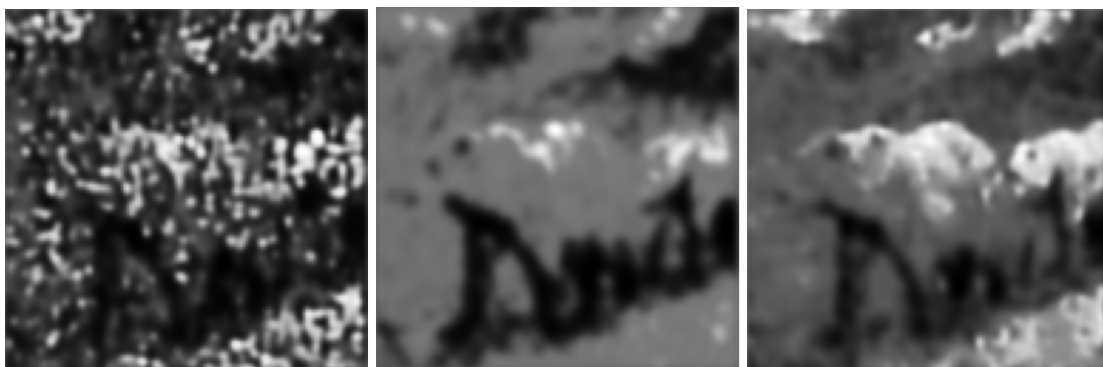


Figura 7. Resultados de filtragem. Da esquerda para a direita: (1) $L = 1$, 20 imagens, 1 época; (2) $L = 5$, 100 imagens, 10 épocas; (3) $L = 5$, 200 imagens, 100 épocas.

5.6.4. Análise Visual

A inspeção visual das imagens evidencia que a presença de ruído speckle é mais pronunciada quando $L = 1$, tornando o processo de filtragem mais desafiador, especialmente para preservar bordas e texturas. Em contrapartida, imagens com $L = 5$ apresentam um ruído menos intenso, permitindo melhores resultados na filtragem.

Além disso, observa-se que o desempenho do modelo melhora à medida que aumenta o número de imagens de treinamento e o número de épocas. Essa tendência pode ser claramente identificada na comparação presente na Figura 7, onde o modelo treinado

com 200 imagens e 100 épocas apresentou os melhores resultados.

Por outro lado, quando o treinamento é realizado com um conjunto reduzido de imagens e poucas épocas o modelo demonstra dificuldades para generalizar, resultando em filtragens incompletas e na perda de detalhes estruturais.

Esses resultados reforçam os desafios envolvidos na filtragem do ruído speckle em imagens *single-look* e evidenciam a importância de definir parâmetros adequados de treinamento, de forma a equilibrar a redução do ruído com a preservação das características estruturais da imagem.

5.7. Discussão dos Resultados

Os resultados apresentados nas análises quantitativas e qualitativas demonstram a eficácia do modelo proposto na filtragem de ruído speckle em imagens SAR, além de destacar os desafios relacionados à preservação das características estruturais das imagens, especialmente em cenários com condições adversas, como imagens *single-look*.

5.7.1. Impacto dos Parâmetros no Desempenho

Os experimentos mostraram que os parâmetros de treinamento (número de imagens utilizadas, número de épocas e número de *looks*) influenciam diretamente no desempenho do modelo:

- Número de *Looks*: Observou-se que um maior número de *looks* ($L = 5$) reduziu significativamente o impacto do ruído speckle, facilitando o trabalho do modelo e permitindo maior preservação de bordas e texturas. Por outro lado, imagens *single-look* ($L = 1$) apresentaram maior granularidade no ruído, dificultando a filtragem e resultando em menor PSNR e SSIM.
- Número de Imagens de Treinamento: Aumentar a quantidade de imagens utilizadas no treinamento contribuiu para melhorar o desempenho do modelo, pois mais dados forneceram maior variabilidade e informações para que a rede generalizasse melhor.
- Número de Épocas: A performance também melhorou com o aumento no número de épocas, especialmente em condições com poucos dados iniciais. No entanto, em configurações com 100 épocas e muitas imagens, os ganhos adicionais foram

marginais, sugerindo que o modelo atingiu um limite de aprendizado, possivelmente relacionado ao fenômeno de *overfitting*.

5.7.2. Comparação com Trabalhos Relacionados

Para contextualizar os resultados obtidos, realizamos uma comparação com os valores apresentados por Penna & Mascarenhas (2019), que estudaram métodos de filtragem de ruído speckle baseados em Non-Local Means e distâncias estocásticas. Embora as imagens utilizadas não sejam as mesmas, os valores médios de PSNR e SSIM obtidos são comparáveis.

De acordo com os resultados apresentados por (Penna & Mascarenhas 2019), os valores médios de PSNR variam entre 21.39 dB e 22.28 dB, enquanto os valores médios de SSIM giram em torno de 0.50 a 0.52, dependendo da distância estocástica utilizada na avaliação do modelo.

No presente estudo, os melhores resultados foram obtidos para $L = 5$, 200 imagens de treinamento e 100 épocas, atingindo $\text{PSNR} = 21.81$ dB e $\text{SSIM} = 0.51$, o que demonstra que o modelo baseado em redes neurais convolucionais foi capaz de alcançar um desempenho semelhante ao obtido pelos métodos baseados em distâncias estocásticas.

Essa comparação sugere que, apesar das diferenças metodológicas e das bases de dados utilizadas, o modelo proposto é competitivo em relação a abordagens estatísticas bem estabelecidas. Além disso, reforça o potencial da abordagem baseada em aprendizado profundo para aplicações na filtragem de ruído speckle em imagens SAR.

5.7.3. Limitações do Modelo e Considerações Finais

Os resultados mostram que é possível desenvolver bons modelos de filtragem de ruído speckle utilizando equipamentos domésticos, sem a necessidade de hardware especializado. O treinamento do modelo foi realizado em um computador pessoal, com processador Intel Core i7, GPU AMD Radeon HD 7400M e 8GB de RAM, demonstrando que redes neurais convolucionais podem ser aplicadas mesmo em ambientes computacionais limitados.

Apesar disso, algumas limitações foram identificadas. A filtragem de imagens *single-look* ainda representa um grande desafio, uma vez que o ruído é significativa-

mente mais intenso e granular. Além disso, o fenômeno de *overfitting* foi observado em configurações com 100 épocas e grandes volumes de dados de treinamento, onde os ganhos marginais de PSNR e SSIM foram reduzidos. Isso indica que o modelo pode ter aprendido padrões específicos do conjunto de treinamento sem melhorar sua generalização para novos dados.

Embora os valores obtidos sejam comparáveis aos da literatura, uma validação adicional com imagens SAR reais ainda se faz necessária para consolidar a aplicabilidade do modelo.

Esses pontos reforçam a necessidade de ajustes adicionais na arquitetura do modelo e no processo de treinamento para otimizar sua capacidade de generalização e preservar detalhes estruturais em imagens *single-look*.

6. Conclusão

Este trabalho apresentou um estudo sobre a filtragem de ruído speckle em imagens SAR utilizando redes neurais convolucionais. Os experimentos demonstraram que o modelo proposto foi capaz de reduzir o ruído speckle, preservando características estruturais das imagens em diferentes condições de treinamento e níveis de ruído. Os resultados quantitativos indicaram que o desempenho do modelo variou de acordo com os parâmetros testados. O aumento no número de *looks* contribuiu para a melhora na qualidade das imagens filtradas, tornando a remoção de ruído mais eficaz. O aumento na quantidade de imagens de treinamento e no número de épocas também proporcionou melhorias, embora os ganhos tenham sido reduzidos quando o número de épocas foi muito elevado, indicando um possível *overfitting*. A filtragem de imagens *single-look* continua sendo um desafio significativo devido à elevada granularidade do ruído speckle.

A inspeção visual confirmou essas tendências, mostrando que imagens com $L = 5$ apresentaram melhores resultados em comparação com imagens *single-look*. Além disso, foi possível identificar que diferentes configurações de treinamento influenciaram na suavização do ruído e na preservação de detalhes estruturais. A comparação com trabalhos anteriores indicou que o modelo baseado em redes neurais convolucionais apresentou desempenho semelhante ao método baseado em Non-Local Means e distâncias estocásticas proposto por Penna e Mascarenhas (Penna & Mascarenhas 2019). Apesar das diferenças metodológicas e das bases de dados utilizadas, os valores médios de PSNR e SSIM obtidos foram comparáveis, sugerindo que abordagens baseadas em aprendizado profundo podem ser competitivas com métodos tradicionais para remoção de speckle.

Um aspecto importante deste estudo foi a viabilidade da implementação do modelo em hardware doméstico, sem a necessidade de infraestrutura computacional avançada. O treinamento foi realizado em um computador pessoal com processador Intel Core i7, GPU AMD Radeon HD 7400M e 8GB de RAM, demonstrando que redes neurais convolucionais podem ser aplicadas mesmo em ambientes computacionais limitados. Embora o tempo de treinamento tenha sido maior do que em sistemas com GPUs de alto desempenho, os resultados indicam que é possível desenvolver modelos eficazes utilizando equipamentos acessíveis.

Além disso, todas as etapas do desenvolvimento foram realizadas utilizando ferramentas de código aberto e gratuitas, como Python 3.9, PyTorch, NumPy, SciPy e OpenCV. O uso dessas ferramentas reduz custos, facilita a reprodutibilidade dos experimentos e promove a colaboração com a comunidade científica. A combinação de hardware acessível e software livre abre novas possibilidades para que pesquisadores e desenvolvedores possam explorar técnicas de filtragem de ruído speckle sem a necessidade de investimentos em infraestrutura computacional de alto custo.

6.1. Trabalhos Futuros

Embora os resultados obtidos tenham sido satisfatórios, há diversas possibilidades para aprimoramento deste trabalho. Uma das direções futuras é a comparação do modelo desenvolvido com outros filtros tradicionais que não utilizam aprendizado profundo, como filtros baseados em estatísticas locais e métodos de filtragem adaptativa. Essa comparação permitirá avaliar melhor as vantagens e limitações da abordagem com redes neurais convolucionais.

Outra questão relevante a ser explorada é a elaboração de estratégias para mitigar problemas de *overfitting* e *underfitting*. O ajuste fino dos hiperparâmetros do modelo, o uso de técnicas de regularização e a experimentação com diferentes arquiteturas de redes neurais podem contribuir para melhorar a generalização do modelo, reduzindo a perda de desempenho quando aplicado a novos conjuntos de dados.

Além disso, a utilização de imagens SAR reais é um passo essencial para validar o modelo em aplicações práticas. Embora os experimentos tenham sido realizados com imagens sintéticas contaminadas com ruído speckle simulado, testar a abordagem em imagens reais permitirá avaliar sua eficácia em cenários mais próximos das aplicações operacionais.

Por fim, propõe-se a elaboração de um banco de dados padronizado com imagens SAR reais, que possa ser utilizado para treinar e avaliar diferentes métodos de filtragem de ruído speckle. A criação de um conjunto de dados bem definido e acessível à comunidade científica pode contribuir significativamente para a padronização das comparações entre diferentes técnicas e para o avanço das pesquisas na área.

Referências

- Arbelaez, P., Maire, M., Fowlkes, C. & Malik, J. (2011). GitHub - BIDS/BSDS500: Berkeley Segmentation Data Set, <https://github.com/BIDS/BSDS500>. [Accessed 26-01-2025].
- Arbeláez, P., Maire, M., Fowlkes, C. & Malik, J. (2011). Contour detection and hierarchical image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33**(5): 898–916.
- Archana, R. & Jeevaraj, P. S. E. (2024). Deep learning models for digital image processing: a review, *Artificial Intelligence Review* **57**(1).
- Argenti, F., Lapini, A., Bianchi, T. & Alparone, L. (2013). A tutorial on speckle reduction in synthetic aperture radar images, *IEEE Geoscience and Remote Sensing Magazine* **1**(3): 6–35.
- Biolchini, J., Mian, P. G., Natali, A. C. C. & Travassos, G. H. (2005). Systematic review in software engineering, *System engineering and computer science department COP-PE/UFRJ, Technical Report ES*.
- Buades, A., Coll, B. & Morel, J. M. (2005). A review of image denoising algorithms, with a new one, *Multiscale Modeling and Simulation* **4**(2): 490–530.
- Cozzolino, D., Verdoliva, L., Scarpa, G. & Poggi, G. (2020). Nonlocal cnn sar image despeckling, *Remote Sensing* **12**(6): 1006.
- Dalsasso, E., Yang, X., Denis, L., Tupin, F. & Yang, W. (2020). Sar image despeckling by deep neural networks: from a pre-trained model to an end-to-end training strategy, *Remote Sensing* **12**(16): 2636.
- Dhruv, A. J., Patel, R. & Doshi, N. (2020). Python: The most advanced programming language for computer science applications, *Proceedings of the International Conference on Culture Heritage, Education, Sustainable Tourism, and Innovation Technologies*.
- Donoho, D. L. & Johnstone, I. M. (1995). Adapting to unknown smoothness via wavelet shrinkage, *Journal of the American Statistical Association* **90**(432): 1200–1224.
- Dutra, L. V., Mura, J. C., da Costa Freitas, C., dos Santos, J. R. & Elmiro, M. T. (2003).

- Processamento de imagens de radar de abertura sintética – princípios e aplicações, *Anais do IV Workshop em Tratamento de Imagens, NPDI/DCC/ICEx/UFMG* pp. 4–13.
- Fielding, E. (2017). Introduction to (SAR) interferometry. Applied Remote Sensing Training Program (ARSET), NASA.
- Fracastoro, G., Magli, E., Poggi, G., Scarpa, G., Valsesia, D. & Verdoliva, L. (2021). Deep learning methods for synthetic aperture radar image despeckling: An overview of trends and perspectives, *IEEE Geoscience and Remote Sensing Magazine* **9**(2): 29–51.
- Frost, V. S., Stiles, J. A., Shanmugan, K. S. & Holtzman, J. C. (1982). A model for radar images and its application to adaptive digital filtering of multiplicative noise, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-4**(2): 157–166.
- Glorot, X., Bordes, A. & Bengio, Y. (2011). Deep sparse rectifier neural networks, in G. Gordon, D. Dunson & M. Dudík (eds), *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, Vol. 15 of *Proceedings of Machine Learning Research*, PMLR, Fort Lauderdale, FL, USA, pp. 315–323.
- Gonzalez, R. C. & Woods, R. E. (2017). *Digital Image Processing*, 4 edn, Pearson, Upper Saddle River, NJ.
- Herm, L.-V., Heinrich, K., Wanner, J. & Janiesch, C. (2023). Stop ordering machine learning algorithms by their explainability! a user-centered investigation of performance and explainability, *International Journal of Information Management* **69**: 102538.
- Jansing, E. D. (2021). *Introduction to synthetic aperture radar: Concepts and practice*, McGraw-Hill Education, Columbus, OH.
- Kingma, D. P. & Ba, J. (2015). Adam: A method for stochastic optimization, in Y. Bengio & Y. LeCun (eds), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.
- Kitchenham, B. A. & Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering, *Technical Report EBSE 2007-001*, Keele University and Durham University Joint Report.

- Kuan, D. T., Sawchuk, A. A., Strand, T. C. & Chavel, P. (1985). Adaptive noise smoothing filter for images with signal-dependent noise, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-7**(2): 165–177.
- Lee, J.-S. (1980). Digital image enhancement and noise filtering by use of local statistics, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-2**(2): 165–168.
- Liu, J. & Liu, R. (2022). Synthetic aperture radar image despeckling using convolutional neural networks in wavelet domain, *IET Image Processing* **17**(9): 2561–2574.
- Loizou, C. P. & Pattichis, C. S. (2015). *Despeckle filtering for ultrasound imaging and video, volume I*, Synthesis lectures on algorithms and software in engineering, Springer International Publishing, Cham.
- Lopes, A., Nezry, E., Touzi, R. & Laur, H. (1990). Maximum a posteriori speckle filtering and first order texture models in sar images, *10th Annual International Symposium on Geoscience and Remote Sensing*, IEEE, p. 2409–2412.
- Michigan Tech Volcanology (1994). Kliuchevskoi volcano optical/radar comparison. Disponível em: <https://www.geo.mtu.edu/volcanoes/klyuchevskoi/images/kliuch-compare.html>. Acesso em: 20 out. 2024.
- Nath, A. (2013). Image denoising algorithms: A comparative study of different filtration approaches used in image restoration, *2013 International Conference on Communication Systems and Network Technologies*, IEEE, p. 157–163.
- Passah, A., Amitab, K. & Kandar, D. (2021). Sar image despeckling using deep cnn, *IET Image Processing* **15**(6): 1285–1297.
- Paszke, A. et al. (2019). Pytorch: An imperative style, high-performance deep learning library, *Advances in Neural Information Processing Systems* 32 pp. 8024–8035.
- Penna, P. A. A. & Mascarenhas, N. D. A. (2019). Sar speckle nonlocal filtering with statistical modeling of haar wavelet coefficients and stochastic distances, *IEEE Transactions on Geoscience and Remote Sensing* **57**(9): 7194–7208.
- Penna, P. A. & Mascarenhas, N. D. (2018). (non-) homomorphic approaches to denoise intensity sar images with non-local means and stochastic distances, *Computers and*

Geosciences **111**: 127–138.

Podest, E. (2024). An introduction to synthetic aperture radar (SAR) and its applications. Applied Remote Sensing Training Program (ARSET), NASA.

PyTorch Contributors (2023). Loading data in pytorch, https://pytorch.org/tutorials/recipes/recipes/loading_data_recipe.html. Accessed: 2025-01-27.

Rudin, L. I., Osher, S. & Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms, *Physica D: Nonlinear Phenomena* **60**(1–4): 259–268.

Santos, C. A. N. (2017). *Redução de ruído speckle em imagens de ultrassom com filtragem não-local e distâncias estocásticas*, Master's thesis, UFSCAR.

Sarkar, R. & Sudeep, P. V. (2024). *A review and implementation of image despeckling methods*, IOP Publishing, pp. 10–1–10–33.

Shamsoddini, A. & Trinder, J. (2010). Image texture preservation in speckle noise suppression, *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives* **38**.

Tian, C., Fei, L., Zheng, W., Xu, Y., Zuo, W. & Lin, C.-W. (2020). Deep learning on image denoising: An overview, *Neural Networks* **131**: 251–275.

Vitale, S., Ferraioli, G. & Pascazio, V. (2021). Multi-objective cnn-based algorithm for sar despeckling, *IEEE Transactions on Geoscience and Remote Sensing* **59**(11): 9336–9349.

Apêndice A – Apresentação do script

Este Apêndice apresenta os detalhes do script desenvolvido para a filtragem de ruído speckle em imagens SAR, incluindo explicações sobre as bibliotecas utilizadas, exemplos de código e resultados obtidos.

O script foi desenvolvido utilizando linguagens e bibliotecas amplamente conhecidas e acessíveis, garantindo sua reprodutibilidade e usabilidade e conta com uma série de funções que estruturam o pipeline para a filtragem de ruído speckle em imagens SAR.

Abaixo, são descritas as funções principais com seus objetivos e trechos de código exemplos.

1. Carregamento de Imagens

A função `load_images_from_folder` é responsável por carregar as imagens do dataset. Ela converte as imagens para escala de cinza e as retorna como uma lista.

```
1 def load_images_from_folder(folder):
2     images = []
3     for filename in os.listdir(folder):
4         img = cv2.imread(os.path.join(folder, filename),
5                           cv2.IMREAD_GRAYSCALE)
6         if img is not None:
7             images.append(img)
8     return images
```

2. Adição de Ruído Speckle

A função `add_speckle_noise` adiciona ruído speckle às imagens limpas, simulando a degradação comum em imagens SAR.

```
1 def add_speckle_noise(image, variance=0.1):
2     row, col = image.shape
3     gamma_noise = np.random.gamma(L, 1.0 / L, (row, col))
4     noisy_image = image * gamma_noise
5     return noisy_image
```

3. Normalização de Distribuições

A função `normalize` ajusta as distribuições para que seus valores sejam adequados às métricas estocásticas e garantindo comparações consistentes entre as distribuições de probabilidade e evitando distorções causadas por escalas ou discrepâncias numéricas.

```
1 def normalize(p, q):
2     p = np.asarray(p).flatten()
3     q = np.asarray(q).flatten()
4     p_sum = np.sum(p)
5     q_sum = np.sum(q)
6     if p_sum > 0:
7         p /= p_sum
8     if q_sum > 0:
9         q /= q_sum
10    return p, q
```

4. Funções de Métricas Estocásticas

O script implementa diversas métricas estocásticas, como a divergência de Kullback-Leibler, que são utilizadas para avaliar a similaridade entre distribuições antes e depois da filtragem.

```
1 def kullback_leibler_divergence(p, q):
2     p, q = normalize(p, q)
3     return entropy(p, q)
4
5 def hellinger_distance(p, q):
6     p, q = normalize(p, q)
7     return euclidean(np.sqrt(p), np.sqrt(q)) / np.sqrt(2)
```

5. Função de Perda Composta

A função de perda combina o erro quadrático médio com métricas estocásticas para melhorar a eficácia da filtragem de ruído, afim de tentar preservar a estrutura es-

tatística da imagem original.

```
1 def combined_loss(output, target, p, q, alpha=0.5, beta=0.3,
2   gamma=0.2):
3     mse_loss = nn.MSELoss()(output, target)
4     kl_loss = kullback_leibler_divergence(p, q)
5     hellinger_loss = hellinger_distance(p, q)
6     js_loss = jensen_shannon_divergence(p, q)
7     total_loss = mse_loss + alpha * kl_loss + beta *
8         hellinger_loss + gamma * js_loss
9     return total_loss
```

6. Treinamento e Avaliação

O treinamento do modelo foi realizado utilizando um pipeline que inclui a criação de pares de imagens ruidosas e limpas, treinamento da rede neural convolucional e validação para monitorar o desempenho.

```
1 def train_model(model, train_loader, val_loader, criterion,
2   optimizer, num_epochs, device):
3     for epoch in range(num_epochs):
4         model.train()
5         train_loss = 0
6         for noisy, clean in tqdm(train_loader, desc=f'Epoch
7             {epoch+1}/{num_epochs}', leave=False):
8             noisy, clean = noisy.to(device), clean.to(device)
9             optimizer.zero_grad()
10            outputs = model(noisy)
11            loss = criterion(outputs, clean)
12            loss.backward()
13            optimizer.step()
14            train_loss += loss.item() * noisy.size(0)
```


7. Exemplo de Saída

Abaixo, apresentamos um exemplo do resultado gerado pelo script, mostrando imagens originais, ruidosas e filtradas lado a lado:

```
1 plt.figure(figsize=(10, 4))
2 plt.subplot(1, 3, 1)
3 plt.title('Original Image')
4 plt.imshow(original_sample.squeeze(), cmap='gray')
5 plt.axis('off')
6
7 plt.subplot(1, 3, 2)
8 plt.title('Noisy Image')
9 plt.imshow(noisy_sample.squeeze().cpu().numpy(), cmap='gray')
10 plt.axis('off')
11
12 plt.subplot(1, 3, 3)
13 plt.title('Denoised Image')
14 plt.imshow(denoised_image.squeeze(), cmap='gray')
15 plt.axis('off')
16 plt.show()
```

Este script, ao ser disponibilizado como material complementar, busca não apenas documentar o trabalho desenvolvido, mas também oferecer uma ferramenta acessível para outros pesquisadores e profissionais interessados na filtragem de ruído speckle em imagens SAR. O projeto completo está disponível em um repositório público no GitHub e pode ser acessado por meio do link:

<https://github.com/pedroarouck/mestrado>